CPUMSX

A REVISTA QUE FAZ O MSX ACONTECER

ANO 3 - N°34 - Cr$ 98.000,00

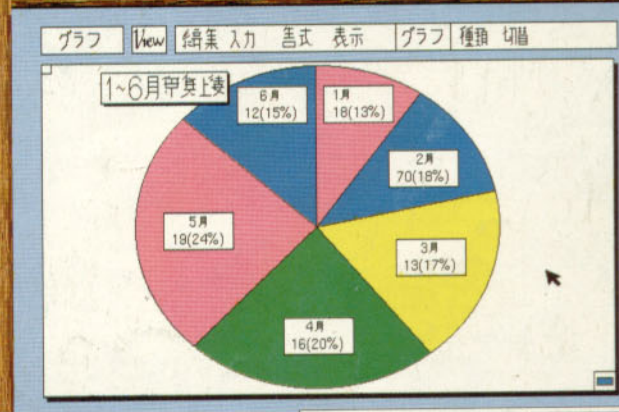
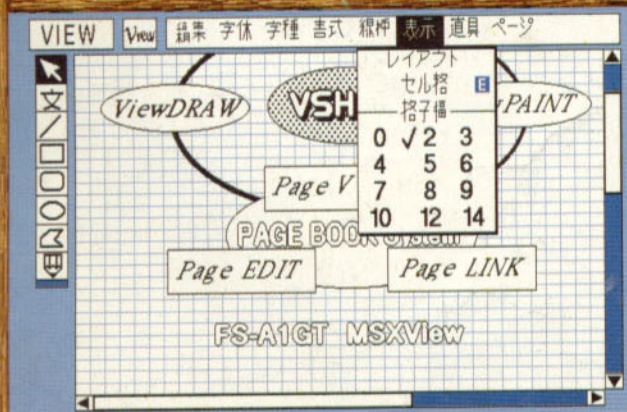
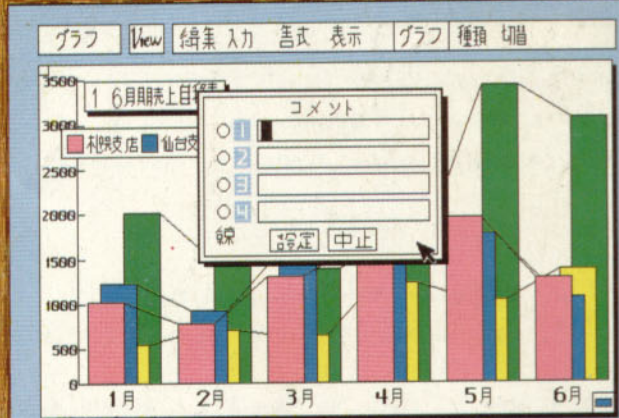
VIA AÉREA: MANAUS, BOA VISTA, SANTARÉM, RIO BRANCO, JIPARANÁ E AMAPÁ

MSX VIEW - O 'WINDOWS' DO MSX

CONSTRUA SUA MEMORY MAPPER - PARTE I

COMPACTADORES: TÉCNICAS E PROGRAMAS

BATALHA 1917: UM FANTÁSTICO JOGO DE ESTRATÉGIA

MAGNAR: A NOVA SAFRA DE JOGOS

SEU PASSAPORTE DE ENTRADA PARA A 7ª FENASOFT CUSTA APENAS US$ 2,00 (*1)

20 a 23 de Julho de 1993
Pavilhão de Exposições
Anhembi - São Paulo

Do dia 1° ao dia 31 de Maio, seu passaporte para a maior Feira de Informática do mundo, (*2) custa apenas US$ 2,00 (câmbio turismo da data de envio do convite). Garanta já o menor preço enviando seu convite ainda este mês. (Veja tabela de datas)

Se você já possui o cartão FENASOFT

O seu cartão 6ª FENASOFT continua válido para a 7ª FENASOFT. Todo visitante que já possui o cartão e quer trocá-lo por qualquer motivo, pagará uma taxa de US$ 5,00 (câmbio turismo do dia do pagamento).

Visitação de Menores

É proibida a entrada de menores de 16 anos, mesmo que acompanhados dos responsáveis ou portando credencial, convite, carta de expositor, etc.

Se não receber, reclame!

Se você não receber seu cartão magnético até o dia 15/06/93, serão aceitas reclamações até o dia 30/06/93.

PREENCHA OS DADOS ABAIXO E ENVIE PARA:
FENASOFT FEIRAS COMERCIAIS LTDA
AV. OSMAR CUNHA, 251 - 9º ANDAR - CENTRO
CEP: 88015-100 - FLORIANÓPOLIS - SC
TEL: (0482) 24.4305 - FAX: (0482) 23.5249

NOME

EMPRESA

ENDEREÇO

CIDADE **UF** **CEP**

PAÍS **TELEFONE**

FAX **DATA DE NASCIMENTO**

1- Qual seu cargo na empresa onde trabalha?

A☐ Presidente
B☐ Vice Presidente
C☐ Sócio
D☐ Gerente Geral
E☐ Diretor/Superintendente
F☐ Gerente de Departamento
G☐ Controller/Tesoureiro
H☐ Consultor/Assessor
I☐ Representante de Vendas
J☐ Analista de Suporte
K☐ Digitador

2- Quantos empregados tem sua empresa?

A☐ mais de 5000
B☐ 1000 a 5000
C☐ 750 a 999
D☐ 500 a 749
E☐ 250 a 499
F☐ 100 a 249
G☐ 75 a 99
H☐ 50 a 74
I☐ 25 a 49
J☐ 10 a 24
K☐ 5 a 9
L☐ menos de 5

3- Qual seu poder de decisão para compras na empresa?

A☐ Autoridade total
B☐ Autoridade alta
C☐ Autoridade limitada
D☐ Recomenda produtos
E☐ Baixo envolvimento em compras
F☐ Nenhum envolvimento em compras

4- Qual o volume de vendas de sua empresa?

A☐ US$ 500 milhões ou mais
B☐ US$ 100 a 499 milhões
C☐ US$ 75 a 99 milhões
D☐ US$ 50 a 74 milhões
E☐ US$ 25 a 49 milhões
F☐ US$ 10 a 24 milhões
G☐ US$ 1 a 9 milhões
H☐ US$ 750 a 999 mil
I☐ US$ 500 a 749 mil
J☐ US$ 250 a 499 mil
K☐ US$ 100 a 249 mil
L☐ menos de US$ 100 mil

5- Qual a principal área de atuação de sua empresa?

A☐ Governamental
B☐ Agricultura, Mineração, Petróleo
C☐ Transportes
D☐ Comunicação
E☐ Manufatura
F☐ Financeira/Contabilidade
G☐ Seguros/Imobiliária
H☐ Médica/Saúde
I☐ Educacional
J☐ Industrial
K☐ Construção Civil
L☐ Fabricante de Software
M☐ Revenda de Software
N☐ Fabricante de Hardware
O☐ Revenda de Hardware
P☐ Fabricante de Periféricos
Q☐ Revenda de Periféricos
R☐ Fabricante de Suprimentos
S☐ Revenda de Suprimentos
T☐ Bureau de Serviços
U☐ Gráfica
V☐ Jornalismo
W☐ Editora

6- Quais as principais aplicações do computador na sua empresa?

A☐ Contabilidade
B☐ Comunicação
C☐ Gerenciamento de Banco de Dados
D☐ Correio eletrônico
E☐ Desk Top/Cad Cam
F☐ Design Gráfico/Multimídia
G☐ Desenvolvimento de aplicações de programas
H☐ Controle de processo/ manufatura
I☐ Científica/Engenharia
J☐ Editor de textos
K☐ Gerenciamento de processos

(*1) Só serão gratuitos os cartões enviados até 30/05, nos anúncios de Promoção das revistas BYTE e CPU.

(*2) Referência da Revista americana Byte, em sua edição de Novembro de 1992.

BÔNUS RIO EDITORA LTDA.
CAIXA POSTAL 11750
CEP 22022-970
RIO DE JANEIRO - RJ
TEL.: (021) 255-4881

DIRETOR EXECUTIVO
JOSE IDEMAR A. NASCIMENTO

JORNALISTA RESPONSÁVEL
DOLAR TANUS
REGISTRO 430-RS

EDITOR TÉCNICO
CARLOS ALBERTO HERSZTERG

CONSULTOR TÉCNICO
JULIO CESAR SILVA MARCHI

ADMINISTRAÇÃO
LUZIMAR GOMES DA SILVA

EDITORAÇÃO ELETRÔNICA E ARTE-FINALIZAÇÃO
JULIO CESAR SILVA MARCHI

REVISÃO
MÁRCIA CHERMAN

PUBLICIDADE
ALEXANDRE MARQUES

ASSINATURAS
LÚCIA HELENA MARCELINO

CAPA
FOCUS INFORMÁTICA

FOTOLITOS
HUNICOLOR

IMPRESSÃO
GRÁFICA LORD

DISTRIBUIÇÃO
FERNANDO CHINAGLIA
DISTRIBUIDORA
R. TEODORO DA SILVA, 907
TEL.: (021) 577-6655

EDITORIAL

Prezados amigos,

A partir desta edição, CPU/MSX volta a ser **a revista do MSX**! Exclusiva!! Sabemos que o período no qual a revista esteve dividida trouxe incertezas para os usuários. Mas tratava-se de uma situação difícil, como podem lembrar. Desse período, entretanto, acredito que a revista tenha saído fortalecida. Tanto é que após cerca de um ano, volta a tratar só de MSX.

Não podemos deixar de enfatizar e agradecer o apoio que recebemos dos leitores. Mesmo aqueles que se sentiram insatisfeitos com o espaço cedido ao Amiga, não deixaram de prestigiar CPU. Sem esse apoio, é claro, a revista deixaria de existir há muito tempo.

Nós, por outro lado, temos tentado renovar o interesse dos usuários, mostrando novas aplicações e produtos para a linha, saindo do estilo "micro decadente", como ocorreu em outras publicações. Afinal, na Europa e no Japão o MSX vai muito bem, obrigado! Só cabe a nós, usuários, manter esse interesse também aqui no Brasil.

Por isso, agora mais do que nunca precisamos do apoio dos leitores. Colaborem com a revista enviando artigos, programas, MSX Bits, dicas, jogos, sugestões...

Mas... aguardem! Estamos com novidades que irão surpreender muita gente. Por enquanto surpreendam-se com CPU/MSX 34!

Carlos Alberto Herszterg

ÍNDICE

MSX OFFLINE LEITOR DE MENSAGENS

Desenvolvido por Fernando da Rocha Carneiro, o MSX OFFLINE versão 1.22 é destinado aos usuários de modens que acessam BBSs. O programa, totalmente escrito em Pascal e Assembler, é um leitor OFFLINE de mensagens extremamente rápido e possui as seguintes característi-cas:

• Leitura de arquivos ".QWK" gerados pelo RAmail ou pelo Mkqwk
• Editor de textos interno, suportando até 99 linhas de 70 caracteres cada, com re-cursos de "word wrap" e "export/import"
• Arquivos de mensagens com até 8 Mb de texto (resposta ou leitura – ".QWK")
• Número ilimitado de mensagens por con-ferência (área)
• Novas "Master Tags", taglines que agora podem ter até 20 linhas de 58 caracteres cada, fáceis de criar e de editar
• Caixa postal do usuário (mensagens en-dereçadas a você)
• Opções de "search" ultra-rápidas, com possibilidade de múltiplas procuras (area, from etc.)
• Gravação automática (opcional) das últi-mas mensagens lidas, o que permite a leitura posterior, mesmo após um "reset"
• Gravação da configuração, que registra o "config" dos drives, além do último banco de taglines usado, que será carregado automaticamente para a memória

Para mais informações, ligue para (021) 393-4490, ou envie uma mensagem para Fernando Carneiro na área MSX da FidoNet (rede nacional de mensagens).

TECNOBYTES DISTRIBUI SOFTS PARA FM

Contando com uma vastíssima biblioteca de músicas para FM, a Tecnobytes está distribuindo gratuitamente parte de seu acervo exclusivamente para software hou-ses. O intuito dessa iniciativa é suprir o mercado, dando opções aos usuários de interfaces FM com músicas inéditas re-tiradas de jogos pela equipe. As empresas interessadas devem enviar três disquetes de 5 1/4, junto com os selos para a remes-sa do material, especificando os dados do equipamento. A Tecnobytes atenderá so-mente as cartas que contenham o CGC e a inscrição estadual da firma.

TECNOBYTES INFORMÁTICA
Caixa Postal 79841
Coelho do Rocha
São João de Meriti – RJ
Cep 25550-970

COBRA SOFTWARE NA EUROPA

A Cobra Software está sendo repre-sentada na Europa pelas seguintes em-presas e pessoas:

• Na Bélgica, em Marcinelle, pela IOD, sob a direção de Olivier Hustin
• Na Bélgica, em Bruxelas, pela revista MSX Forum Magazine, sob a direção de Alexandre Rajszajt
• Na Suíça em Lausane pelo Les MSXiens Suisse, de Carlos Leitão
• Na França, em Domerati, por Faveire Sébastien
• Na França, em Saint Etienne, por André Vermelle

O objetivo, a princípio, é o de exportar softwares e sharewares brasileiros desen-volvidos com esta finalidade. Através deste intercâmbio, a empresa está recebendo quinzenalmente softwares da Bélgica, França, Suíça, Holanda, Japão etc.

FREE SHAREWARES COBRA E CPU

A Cobra Software, com o apoio de revista CPU-MSX, está oferecendo aos leitores dois excelentes programas:

RHAPSODY desenvolvido pela IOD da Bél-gica, para MSX 2.0, FM PAC ou MSX AUDIO CHIP. Se você não possui o FM PAC nem MSX AUDIO CHIP, poderá as-sim mesmo apreciar o demo RHAPSODY em seu MSX 2.0 sem a trilha musical.

LABORATÓRIO DE VÍRUS programa didático para MSX 1.0, onde você poderá familiari-zar-se com a manipulação de vírus para MSX, entre eles: NO DIR, VIRUS MSX, SEXTA FEIRA 13, HELL RAISER e CHU-PETA. Acompanha a versão shareware do ANTI-VÍRUS, versão beta, de Felipe Vie-cente de Azevedo de Albertão.

Para receber gratuitamente os programas, envie um disquete formatado em 40 ou 80 trilhas e cinco selos "tarifa única" para as despesas de correio e em-balagens.

Rogério Bello dos Santos

COBRA SOFTWARE
Rua Chady Muradi, 81
Jaguaré – São Paulo – SP
Cep 05351-050

MSX ATACA NO VIDEOTEXTO

Já está no ar, em sua segunda edição, a primeira revista "on-line" do Brasil para MSX. A MSX ON-LINE, um projeto pionei-ro idealizado por Alexandre Sobrino espe-cialmente para o fornecedor de serviços do Videotexto, UNISANTOS (Universida-de Católica de Santos). O acesso é feito pela central 1481 com a chave PLAY*MSX no menu inicial

A revista, ainda em fase de implantação, aceita colaborações em forma de artigo, dicas e promoverá a distribuição de "demos" internacionais, cedidos por Rogério Bello, da Cobra Soft, consultor e colaborador permanente de revista. Figuram entre as matérias deste periódico quinzenal: análises de software, artigos técnicos sobre o MSX 2.0 e, em breve, com exclusividade, análise das pautas das próximas edições de CPU, a revista do MSX.

Envie suas colaborações para:

MSX ON-LINE
A/C Alexandre Sobrino
Av. Almirante Cóchrane, 287
Embaré – Santos – SP
Cep 11040-003

UM PODEROSO EDITOR GRÁFICO

Ainda sem representação comercial, Charles de Rossi divulga seu editor gráfico com vários recursos, como o "zoom", ani-mações com Sprites e tridimensionais e muito mais.

O autor espera obter contatos para representação e venda do produto.

AUTOR: **Charles de Rossi**
Tel.: (011) 432-2394

FANTASY BBS

Surge mais uma BBS para os usuários do MSX com modem. É a Fantasy BBS, que realmente oferece todo o apoio à linha. A Fantasy BBS possui cerca de 11 Megaby-tes em programas compactados exclusi-vamente para o MSX, menus de 40 colu-nas (para quem possui MSX 1) rápidos e objetivos, além de oferecer áreas especí-ficas de programas para MSX 2 e FM-Sound.

Em funcionamento 24 horas por dia e agora cadastrada pelo SysOp Erick Oliveira na rede internacional FidoNet (node 4:802/31), a Fantasy atende no telefone (021) 553-4453, nas velocidades 300, 1200 e 2400 bps – 8N1.

MSX 2 e MSX 2+

Comandos de hardware

Julio Cesar Silva Marchi
André Luiz Rocha Tupinambá

Desde que iniciamos esta série, citamos várias vezes os comandos de hardware que os novos VDPs possuem. Para esclarecer o que é isso, vamos nos basear em um questionamento: como fazemos para manipular o VDP do MSX 1 em Assembly? Neste caso, possuímos duas opções, o BIOS do micro e os famosos OUTs (exatamente o que o BIOS faz).

Entretanto, que tal se nós lhe pedíssemos para traçar uma reta qualquer em Assembly? Quem já passou por isto sabe exatamente onde estamos querendo chegar. No MSX 1 não existe nenhuma rotina apontada pela tabela de JUMPs do BIOS destinada a essa aplicação (e a muitas outras). Apesar disso, muitos usuários curiosos já descobriram que dentro do interpretador Basic existe uma rotina que é usada pelo comando LINE, destinada exatamente para este fim. Pronto: o problema estaria resolvido. A tal rotina funciona maravilhosamente bem!

Mas para quem pensava que era só ir utilizando esta rotina e tudo bem, mal sabia que estava criado outro problema (ou outros!). Isso porque, segundo as regras divulgadas pela Microsoft, nunca devemos chamar uma rotina que esteja fora da tabela de JUMPs, pois corre-se o risco de não encontrá-la na mesma posição de memória no BIOS dos micros que forem lançados posteriormente. Seu programa só funcionaria então, teoricamente, em micros MSX versão 1.x (e olhe lá!). Como você pôde notar, para nós programadores, as coisas sempre tendem a se complicar cada vez mais, principalmente em uma arquitetura que possui a **compatibilidade** como proposta fundamental de marketing.

Nesse ponto você então pensa na segunda opção – fazer uma rotina que calcule cada ponto da reta e o plote na tela – como a alternativa final. Se você realmente pensou nisto, desculpe-nos por desapontá-lo novamente, mas poucos conseguiram tal façanha com eficiência.

Agora, o que isso tudo tem a ver com o assunto? Tudo, é claro! Foi pensando em solucionar este tipo de inconveniente que os projetistas da Yamaha resolveram incorporar no próprio VDP várias "rotinas" para agilizar e facilitar o trabalho dos programadores (as quais algumas já foram explicadas no artigo anterior).

CÓPIA LÓGICA

Se você entendeu bem o que foi dito até agora, não terá nenhum problema nas aplicações que serão demonstradas adiante. Cópia lógica é exatamente aquela usada pelo novo comando COPY do BASIC (dê uma olhadinha no manual que acompanha o seu micro). Além de copiar parte da tela, esta rotina também faz uma operação lógica entre a cor do pixel que já existe com a cor do pixel que iremos colocar. Repare então que existe uma diferença significativa entre a atuação destes comandos comparados com os de cópia rápida. Nestes, a cópia se processa a partir das cores e não por bytes.

Na figura 5 (CPU/MSX Nº 33 pág. 9) temos uma tabela que mostra as operações lógicas com seus nomes e códigos.

A partir de agora os comandos que deverão ser enviados a CMR serão representados de uma forma diferente, por exemplo: 0BxH, onde "x" se refere ao código da operação lógica desejada que está indicado na figura 5 (CPU/MSX Nº 33 pág. 9). Não se esqueça que nas cópias lógicas cada byte transferido corresponde a um pixel da tela e não a um byte da VRAM como eram nas cópias aceleradas.

Comando LMMC (Transferência CPU - VRAM)

Este comando é semelhante ao comando HMMC. Para executá-lo devemos setar: DX, DY, NX, NY, os bits MXD, DIX, DIY, a cor do primeiro PIXEL em CLR e enviar o comando 0BxH. Depois que executarmos este comando também devemos utilizar aquele algoritmo listado anteriormente para transferência de dados entre a CPU e a VRAM (CPU/MSX Nº 33 pág. 9).

Comando LMCM (Transferência VRAM - CPU)

Este comando é exatamente o inverso do LMMC. Ele faz a cópia de uma área da VRAM para a CPU. Os registradores que devemos setar são: SX, SY, NX, NY, os

bits MXS, DIX, DIY de ARG e enviar o comando 0A0H para CMR. Depois de executado o comando devemos usar aquele algoritmo já mencionado com uma pequena modificação: ao invés de escrevermos um byte na port#3, devemos obter o conteúdo do registrador de status #7, que é onde a cor do pixel retorna para você.

Comando LMMM (Transferência VRAM - VRAM)

Este é outro comando muito usado, ele é a versão lógica para o comando HMMM. Nele devemos setar os seguintes registradores: SX, SY, DX, DY, NX, NY, os bits MXS, MXD, DIX, DIY de ARG e enviar o comando 9xH para CMR.

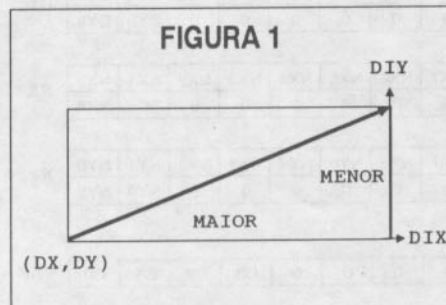
Comando LMMV (Transferência VDP - VRAM)

Mais uma versão lógica para um comando acelerado, HMMV no caso. Para variar, funciona exatamente da mesma maneira só que agora mais lento, com operação lógica e sem aquele recurso especial comentado. Os registradores usados são: DX, DY, NX, NY, os bits MXD, DIX, DIY de ARG e o comando 8xH para CMR.

COMANDOS EXTRAS

Entramos agora nos comandos que proporcionam as maiores vantagens dos comandos de hardware. Com eles podemos fazer coisas que no MSX 1 se fazia a custo de complicados algoritmos (como traçar uma linha por exemplo).

Comando LINE (desenha uma linha)

Este comando veio solucionar grandes problemas que os programadores de assembly tinham quando queriam fazer algo em modo gráfico no MSX 1. Apesar da definição das coordenadas da linha serem um pouco confusas, este comando provavelmente será o primeiro que vocês irão testar (pelo menos foi o que aconteceu conosco). A definição das coordenadas, como já havíamos comentado antes, é um pouco confusa e por isto vamos comentar o estado dos registradores mais detalhadamente, para que as coisas não fiquem muito difíceis.

Analise a figura 1. As coordenadas iniciais devem ser passadas por DX e DY. O tamanho do lado do triângulo indicado como MAIOR deve ir no registrador NX e o outro no NY. A cor vai no registrador CLR e os bits de ARG são: MXD, DIX, DIY e MAJ. O bit MAJ é destinado à definição de qual eixo será considerado como o maior lado indicado em NX. Após todo este trabalho é só enviar o comando 7xH para CMR.

Comando PSET (Seta um pixel na tela)

Este comando é um dos mais simples dos aqui listados e dispensa maiores comentários. Para utilizá-lo basta setar os seguintes registradores: DX, DY, CLR e enviar o comando 5xH para CMR.

Comando POINT (Retorna a cor de um ponto na tela)

Este comando, como vocês já devem ter reconhecido pelo nome, retorna para você a cor de algum ponto da tela. Para utilizá-lo basta setar os registradores: SX, SY, o MXS de ARG e enviar o comando 40H a CMR. Depois de executado, a cor do ponto marcado retorna no registrador de status #7.

Comando SRCH (Procura uma cor na direção X)

Este comando é uma das grandes virtudes do novo VDP. Ele era simulado no MSX 1 para fazer o comando PAINT de uma área da tela. Este comando faz exatamente o seguinte: você marca um ponto na tela e, a partir deste, o VDP procura uma cor à direita ou à esquerda. Para ficar melhor ainda, é possível escolher se desejamos que este comando nos indique se encon-

trou um ponto igual ou diferente ao desejado, isto é, o SEARCH (SRCH) pode ir procurando até encontrar um ponto igual ou então até encontrar um ponto diferente do especificado.

Para usar este comando devemos usar os seguintes registradores: SX, SY, MXD, DIX, EQ de ARG, a cor em CLR e enviar o comando 60H para CMR. Ao final de sua execução, devemos ler o registrador de status #2 e analisar o bit BD (bit 4). Se ele estiver setado, a cor foi encontrada e a coordenada X do ponto estará guardada nos registradores de status #8 e #9.

Comando STOP
(Pára a execução de outros comandos)

Este comando serve simplesmente para forçar a interrupção de qualquer outro comando que o VDP esteja executando. Para utilizá-lo devemos enviar o byte 00 para CMR e pronto, o VDP estará livre para ser usado.

Na figura 2 temos o mapa dos registradores usados pelos comandos de hardware que foram ensinados até então. Nas páginas seguintes está o mapa completo dos registradores do VDP do MSX 2 (como fora prometido). Baseando-se no que já foi dito, será fácil compreender as explicações contidas lá.

Se você tem acompanhado esta série, provavelmente já estará apto a colocar boa parte de suas idéias em prática. Se você já tem algo pronto, mande-nos. Qualquer dúvida, escreva... ❑

FIGURA 2 – Registradores utilizados pelos comandos de hardware

MSB	7	6	5	4	3	2	1	0 LSB		
R#32	SX7	SX6	SX5	SX4	SX3	SX2	SX1	SX0	SX	Utilizado como entrada de coordenadas orígem para todos os comandos de hardware.
R#33	0	0	0	0	0	0	0	SX8		
R#34	SY7	SY6	SY5	SY4	SY3	SY2	SY1	SY0	SY	
R#35	0	0	0	0	0	0	SY9	SY8		
R#36	DX7	DX6	DX5	DX4	DX3	DX2	DX1	DX0	DX	Utilizado como entrada de coordenadas destino para todos os comandos de hardware.
R#37	0	0	0	0	0	0	0	DX8		
R#38	DY7	DY6	DY5	DY4	DY3	DY2	DY1	DY0	DY	
R#39	0	0	0	0	0	0	DY9	DY8		
R#40	NX7	NX6	NX5	NX4	NX3	NX2	NX1	NX0	NX	Largura vertical e horizontal da área a ser manipulada pelos comandos de hardware.
R#41	0	0	0	0	0	0	0	NX8		
R#42	NY7	NY6	NY5	NY4	NY3	NY2	NY1	NY0	NY	
R#43	0	0	0	0	0	0	NY9	NY8		
R#44	0	0	0	0	C3	C2	C1	C0	CLR →	Controla a cor nas SCREENs 5 e 7
R#44	0	0	0	0	0	0	C1	C0	CLR →	Controla a cor na SCREEN 6
R#44	C7	C6	C5	C4	C3	C2	C1	C0	CLR →	Controla a cor na SCREEN 8
R#45	0	0	MXD	MXS	DIY	DIX	EQ	MAJ	ARG →	Registrador usado para passagem dos argumentos para o VDP.
R#46	CM3	CM2	CM1	CM0	LO3	LO2	LO1	LO0	CMR →	Registrador usado para indicar ao VDP qual comando desejamos executar e com qual operação lógica.

Bits de R#45 (ARG):
- MAJ → Indicador do lado maior para comando LINE
- EQ → Condição para término do comando SEARCH
- DIX → Direção no eixo X (0=Para a direita / 1=Para a esquerda)
- DIY → Direção no eixo Y (0=Para baixo / 1=Para cima)
- MXS → Seleção de memória para orígem (0=VRAM / 1=Expanded RAM)
- MXD → Seleção de memória para destino (0=VRAM / 1=Expanded RAM)

FIGURA 3– Mapa de bits dos registradores do MSX 2

- Registradores de modo:

	MSB 7	6	5	4	3	2	1	0 LSB	
R#0	0	DG	IE2	IE1	M5	M4	M3	0	Registrador de Modo 0
R#1	0	BL	IE0	M1	M2	0	SI	MAG	Registrador de Modo 1
R#8	MS	LP	TP	CB	VR	0	SPD	BW	Registrador de Modo 2
R#9	LN	0	S1	S0	IL	E0	*NT	DC	Registrador de Modo 3

* Indicador de negação lógico

R#0:
- Bit 0 – Sempre 0;
- Bit 1 – Usado para seleção do modo (M3);
- Bit 2 – Usado para seleção do modo (M4);
- Bit 3 – Usado para seleção do modo (M5);
- Bit 4 – Habilita a interrupção de *Scanning Line* horizontal (IE1);
- Bit 5 – Habilita a interrupção do *lightpen* (IE2);
- Bit 6 – Seta o *Color Bus* como entrada e coloca os dados na VRAM (DG);
- Bit 7 – Sempre 0;

R#1:
- Bit 0 – Modo do Sprite (MAG : 0=Normal / 1=Estendido);
- Bit 1 – Tamanho do Sprite (SI : 0=8x8 / 1=16x16);
- Bit 2 – Sempre 0;
- Bit 3 – Usado para seleção do modo (M2);
- Bit 4 – Usado para seleção do modo (M1);
- Bit 5 – Habilita a interrupção de *Scanning Line* horizontal (IE0);
- Bit 6 – Usado para ativar/desativar a tela (BL : 0=inativa / 1=ativa);
- Bit 7 – Sempre 0;

R#8:
- Bit 0 – Seleciona visualização em colorido/tons de cinza (BW : 0=colorido / 1=PB);
- Bit 1 – Desabilita od Sprites (SPD);
- Bit 2 – Sempre 0;
- Bit 3 – Seleciona o tipo de VRAM (VR : 0=16K por 1 ou 4 bits / 1=64K por 1 ou 4 bits);
- Bit 4 – Seta *Color Bus* para entrada (CB : 0=Saída / 1=Entrada);
- Bit 5 – Controle da cor 0 (TP : 0=Transparente / 1=Palette);
- Bit 6 – Habilita a *Lightpen* (LP);
- Bit 7 – Controle de *Color Bus* e habilitação do mouse (MS : 0=Bus como saída e mouse inativo / 1=Bus como entrada e mouse ativo);

R#9:
- Bit 0 – Quado 1, seta DLCLK para entrada. Quando 0 seta DLCLK para saída (DC);
- Bit 1 – Quando 1, seleciona o modo PALM (313 linhas). Quando 0, seleciona o modo NTSC (262 linhas). Ambos para saída RGB (*NT);
- Bit 2 – Quando 1, mostra duas telas gráficas intercambiadas no *interlace* (424 pontos), quando 0, mostra apenas uma tela (212 pontos) - (E0);
- Bit 3 – Quando 1, *interlace* ligado. Quando 0, *interlace* desligado;
- Bit 4 – Seleciona modos simultâneos (S0);
- Bit 5 – Seleciona modos simultâneos (S1);
- Bit 6 – Sempre 0;
- Bit 7 – Quando 1, seta o contador horizontal de pontos para 212. Quando 0, seta o contador horizontal de pontos para 192 (LN);

- Registradores de endereço:

	MSB 7	6	5	4	3	2	1	0 LSB	
R#2	0	A16	A15	A14	A13	A12	A11	A10	Tabela de nomes
R#3	A13	A12	A11	A10	A9	A8	A7	A6	Tabela de cores (LOW)
R#10	0	0	0	0	0	A16	A15	A14	Tabela de cores (HIGH)
R#4	0	0	A16	A15	A14	A13	A12	A11	Tabela de padrões
R#5	A14	A13	A12	A11	A10	A9	A8	A7	Atrib. dos Sprites (LOW)
R#11	0	0	0	0	0	0	A16	A15	Atrib. dos Sprites (HIGH)
R#6	0	0	A16	A15	A14	A13	A12	A11	Formação dos Sprites

- Registradores de cor:

	MSB 7	6	5	4	3	2	1	0 LSB	
R#7	TC3	TC2	TC1	TC0	BD3	BD2	BD1	BD0	Cor do texto
R#12	T23	T22	T21	T20	BC3	BC2	BC1	BC0	Cor do Blink
R#13	ON3	ON2	ON1	ON0	OF3	OF2	OF1	OF0	Período de Blink
R#20	0	0	0	0	0	0	0	0	Color Burst 1
R#21	0	0	1	1	1	0	1	1	Color Burst 2
R#22	0	0	0	0	0	1	0	1	Color Burst 3

- TC3 até TC0: Especifica a cor do texto nos modos TEXT-1 e TEXT-2;
- BD3 até BD0: Nos modos TEXT-1 e TEXT-2 especifica a cor de fundo, nos outros modos especifica a cor da borda;
- T23 até T20: Especifica a cor da frente do Blink;
- BC3 até BC0: Especifica a cor de fundo do Blink;
- ON3 até ON0: Especifica o período do Blink aceso;
- OF3 até OF0: Especifica o período do Blink apagado;

Obs: Se zerarmos todos os bits dos registradores R#20, R#21 e R#22, o sinal de *color burst* da saída de vídeo composto será apagado.

- Registros de controle de vídeo:

	MSB 7	6	5	4	3	2	1	0 LSB	
R#18	V3	V2	V1	V0	H3	H2	H1	H0	Ajuste de tela
R#19	IL7	IL6	IL5	IL4	IL3	IL2	IL1	IL0	*Interrupt Line*
R#23	DO7	DO6	DO5	DO4	DO3	DO2	DO1	DO0	SCROLL

- V3 até V0: Controle vertical de vídeo - usado pelo SET ADJUST(x,y);
- H3 até H0: Controle horizontal de vídeo - usado pelo SET ADJUST(x,y);
- IL7 até IL0: Especifica a linha de interrupção do *scanning line*;
- DO7 até DO0: Define a primeira linha da tela - usado pelo SET SCROLL do MSX 2+;

- Registradores de acesso:

	MSB 7	6	5	4	3	2	1	0 LSB	
R#14	0	0	0	0	0	A16	A15	A14	Página da VRAM
R#15	0	0	0	0	S3	S2	S1	S0	Ponteiro para S#n
R#16	0	0	0	0	C3	C2	C1	C0	Ponteiro para P#n
R#17	AII	0	RS5	RS4	RS3	RS2	RS1	RS0	Ponteiro para R#n

- R#14 : Seleciona a página da VRAM / Extended RAM a ser acessada;
- R#15 : Seleciona qual o registrador de Status será lido;
- R#16 : Indica qual Palette será acessada. Para tanto, coloca-se o número da Palette no R#16, em seguida, envie pela Port#2 os níveis de vermelho e azul (os bits de 0 a 2 contém o azul, os bits de 4 a 6 o vermelho e o bit 7 não tem significado). Em seguida envie o nível de verde também pela Port#2 (bits de 0 a 2). Observe que o valor registrador 16 é auto-incrementável, isto significa que se desjarmos acessar a próxima Palette, basta apenas enviar os dados da cor para ela da mesma forma que foi feito no primeiro acesso (primeiro os níveis de azul e vermelho e depois o nível de verde pela Port#2).
- R#17 :
  - RS5 até RS0: Seleciona qual o registrador será escrito (método conhecido como acesso indireto);
  - AII: Indica o auto-incremento deste registrador. Quando 1, está desabilitado e quando 0 está habilitado.

FIGURA 3– Mapa de bits dos registradores do MSX 2 (continuação)

- Registradores de Status:

	MSB 7	6	5	4	3	2	1	0 LSB	
S#0	F	5S	C	Número do 5º Sprite					Equivale à VDP(8)
S#1	FL	LPS	Identificação do MSX-VIDEO					FH	Equivale à VDP(-1)
S#2	TR	VR	HR	BD	1	1	EO	CE	Equivale à VDP(-2)
S#3	X7	X6	X5	X4	X3	X2	X1	X0	Equivale à VDP(-3)
S#4	1	1	1	1	1	1	1	X8	Equivale à VDP(-4)
S#5	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0	Equivale à VDP(-5)
S#6	1	1	1	1	1	1	Y9	Y8	Equivale à VDP(-6)
S#7	C7	C6	C5	C4	C3	C2	C1	C0	Equivale à VDP(-7)
S#8	BX7	BX6	BX5	BX4	BX3	BX2	BX1	BX0	Equivale à VDP(-8)
S#9	1	1	1	1	1	1	1	BX8	Equivale à VDP(-9)

S#0: Bits de 0 a 4 – Usado para indicar o número do Sprite excedente;
Bit 5 – Indicador de colisão de Sprites (C);
Bit 6 – Usado para indicar a existência de Sprites excedentes (5S);
Bit 7 – Indicador de interrupção. A cada interrupção do VDP este bit é setado. Este bit é analisado principalmente pela rotina de interpretação de teclado para saber se foi chamada por uma interrupção do VDP do tipo IE0 (F);

S#1: Bit 0 – Indicador de interrupção do tipo IE1 (FH);
Bits de 1 a 5 – Número de identificação do MSX-VIDEO (ID#);
Bit 6 – Se a *lightpen* estiver ativa, este bit controla se o contato está ativo ou não. Se o mouse estiver ativo, este controla o botão 1 (LPS);
Bit 7 – Se a *lightpen* estiver recebendo sinal luminoso, este bit será comparado com IE2 e se ambos estiverem setados será enviada uma ordem para habilitar a interrupção de leitura para a *lightpen* (FL).

S#2: Bit 0 – Quando 1, o VDP está executando um comando de hardware. Quando 0, o VDP está livre (CE);
Bit 1 – Indica em qual dos bancos de 64Kb da VRAM o VDP deve trabalhar (EO: 0=banco 0, 1=banco 1);
Bit 2 – Sempre 1;
Bit 3 – Sempre 1;
Bit 4 – Indicador de resultado do comando SRCH, (BD: 0=pedido não encontrado, 1=pedido encontrado);
Bit 5 – Durante a interrupção IE1 este bit está setado (HR);
Bit 6 – Durante a interrupção IE0 este bit está setado (VR);
Bit 7 – Quando este bit está setado, indica que o VDP está pronto para transferência de dados (TR: usado pelos comandos de hardware);

S#3: Quando a *lightpen* estiver ativa, este registrador guarda a parte baixa (LOW) da coordenada X do último ponto detectado. Quando o mouse estiver ativo, este indica a parte baixa da coordenada X atual deste;

S#4: Quando a *lightpen* estiver ativa, este registrador guarda a parte alta (HIGH) da coordenada X do último ponto detectado. Quando o mouse estiver ativo, este indica a parte alta da coordenada X atual deste;

S#5: Quando a *lightpen* estiver ativa, este registrador guarda a parte baixa (LOW) da coordenada Y do último ponto detectado. Quando o mouse estiver ativo, este indica a parte baixa da coordenada Y atual deste;

S#6: Quando a *lightpen* estiver ativa, este registrador guarda a parte alta (HIGH) da coordenada Y do último ponto detectado. Quando o mouse estiver ativo, este indica a parte alta da coordenada Y atual deste;

S#7: Este registrador serve apenas para o comando POINT. Nele retorna a cor do ponto selecionado;

S#8: Registrador de uso exclusivo do comando SRCH. Nele retorna a parte baixa (LOW) da coordenada X da cor encontrada;

S#9: Registrador também de uso exclusivo do comando SRCH. Nele retorna a parte alta (HIGH) da coordenada X da cor encontrada;

Compactadores

Técnicas e programas

Miguel Freitas

INTRODUÇÃO

A crescente necessidade de se armazenar informações nos computadores faz com que constantemente sejam desenvolvidos novos dispositivos de armazenamento de dados. Acompanhando esta evolução, os programadores sempre desenvolvem novos programas que conseguem economizar espaço nos discos. Esses "milagrosos" utilitários são os chamados compactadores.

Quem pensa que um disquete de 720 Kbytes só pode conter 720 Kbytes está enganado... Quero dizer, em parte! Sempre há uma maneira de tornar este espaço "maior". Existem programas que conseguem colocar dois, três e até quatro discos em apenas um, utilizando vários macetes.

Se você acha que isso é privilégio dos usuários de outras máquinas como o PC, vou mostrar que não é verdade. Embora seja uma tarefa que precise de memória e velocidade, o MSX pode compactar e descompactar dados, sim!

Atualmente já existem vários compactadores para o MSX, alguns deles criados para o MSX, como o Master Cruncher, e outros trazidos do CP/M. Os usuários de modem utilizam muito estes programas como uma forma de diminuir o tamanho dos arquivos, tornando a ligação telefônica menor e conseqüentemente mais barata.

Vou tentar explicar um pouco da teoria dessas técnicas.

Run-Length

Compactação por repetição. Nesta técnica, que é a mais simples, os dados são compactados quando existe a repetição de bytes, ou seja, se o programa encontra vinte letras "A" em um texto, ele pode substituir isto por algo como "20A". Quando o programa estiver descompactando o arquivo, para que este possa ser lido de maneira normal, ao encontrar um código "20A" ele saberá que no arquivo final isso deverá ser substituído por vinte letras "A". Esta técnica de compactação, embora extremamente simples, é muito fácil de ser implementada e pode ser muito útil na compactação de desenhos que possuem muitas repetições de cores. Este método já foi utilizado nos Winchesters de PC (os RLL) para aumentar a capacidade de armazenamento.

Lempel-Ziv-Welch (LZW)

Esta técnica, desenvolvida por três pessoas, é muito utilizada atualmente, conseguindo um desempenho muito melhor que a Run-length. Aqui, a compactação é feita com a repetição de pedaços em um arquivo. Vejamos o exemplo deste texto:

"A voz preguiçosa, com sua inflexão ligeiramente irônica, despertou muitas lembranças de Van Aldin. Ele tinha charme – sempre tivera charme. Ele observou atentamente o genro." (Agatha Christie)

poderia ficar:

"A voz preguiçosa, com sua inflexão ligeiramente irônica, despertou m*tas *mbranç* * V* Aldin. E* tinha char* – sem* *ve* *. * obs*u at*a* o genro."

No lugar dos asteriscos o compactador colocaria algum código para informar algo como "este trecho já existe na tabela no lugar X", economizando desta forma vários bytes, sem perder as informações.

Como você deve ter percebido, quanto maior é o texto, melhor pode ser a compactação com esta técnica. À medida que o compactador vai lendo o arquivo, vai montando uma tabela com o que ele já conhece e sempre procura se os novos dados existem nesta tabela. Com um arquivo grande, o compactador terá um "vocabulário" de trechos do arquivo bem rico. Esta técnica é utilizada, por exemplo, nas telas gráficas ".GIF" do PC.

Huffman

Esta técnica também é muito utilizada atualmente nos mais diversos compactadores. Embora seja uma técnica de "dois passos" – ou seja, há a necessidade do arquivo ser lido duas vezes –, esta técnica pode ter um desempenho muito bom.

A técnica consiste em criar seqüências de bits de tamanho variável no lugar dos códigos normais de 8 bits. Se o arquivo a ser compactado só possuir códigos ASCII, cada byte ocuparia apenas 7 dos 8 bits, possibilitando uma economia real de 12.5%. Se você quiser utilizar apenas letras maiúsculas no arquivo, você usaria cerca de 100 dos códigos ASCII de 0 a 255. Assim pode-se economizar 40% do arquivo, caso os bits não utilizados forem aproveitados por outros caracteres. O Huffman pode, por exemplo, armazenar a letra "a" (a mais utilizada na língua portuguesa) em apenas um bit: "1", economi-

zando assim 87.5% do espaço normal da letra "a".

Para decidir como proceder a compactação, o compactador vai primeiro ler todo o arquivo, contando quais são os bytes mais usados. Após isto, ele vai montar uma espécie de tabela com os códigos binários de cada byte. Esta tabela será gravada no arquivo destino para uma posterior descompactação. Então o compactador voltará ao início do arquivo e cada byte novo será transformado em uma seqüência de bits com um tamanho variável, que são gravados no arquivo.

Pode acontecer que um arquivo fique maior do que era. Isto ocorre quando o arquivo original usa muitos bytes diferentes e com poucas diferenças entre as freqüências de bytes.

O QUE EXISTE PARA O MSX

Podemos separar os compactadores em dois tipos: os que geram arquivos autodescompactáveis e os que compactam o arquivo para armazenamento que, para ser utilizado, deve primeiro ser descompactado com outro programa para este fim.

O primeiro tipo é amplamente utilizado nos micros IBM PCs, em programas como o Stacker que deixam todos os dados compactados no disco e qualquer acesso que seja feito a ele passa por um compactador ou por um descompactador. Este tipo de programa pode gerar alguns problemas para o usuário, mesmo no PC onde os programas costumam utilizar as rotinas do DOS para acessar o drive ou o HD. No MSX, onde cada programador faz sua própria rotina de acesso ao drive, indo diretamente no hardware, este tipo de compactador nunca funcionaria para todos os programas. Por isso, só conheço dois programas que fazem algo do gênero: o Master Cruncher e o PMARC que falarei adiante.

O Master Cruncher foi desenvolvido para o MSX e possui toda uma apresentação gráfica que o micro merece. Não obtive informações quanto ao método utilizado na compactação, mas a julgar pelo baixo desempenho na compactação de textos, imagino que seja o Run-Length. Ele possui opções para compactar e descompactar arquivos e é acompanhado também de dois módulos ".BIN" e ".COM" que, se carregados antes do programa, podem descompactá-lo automaticamente quando for executado. Este programa é muito útil com telas gráficas (como as do Graphos 3) que podem ser chamadas pelo Basic, ocupando menos espaço em disco, mas sem perder as imagens. Não incluí os resultados dos testes com o Master Cruncher porque além das taxas de compactação serem muito baixas, ele não funcionou corretamente no meu drive de 720 Kbytes.

Entre os compactadores do outro tipo, no qual os dados precisam ser descompactados por um programa antes de serem utilizados, existem diversas opções para o MSX.

SQUEEZE

Este compactador, trazido do CP/M, utiliza o método Huffman. Apesar de não apresentar taxas de compactação muito boas, os tempos de compactação e de descompactação ficaram entre os melhores. Parece ser o mais antigo compactador para CP/M – foi feito em 1984.

Os arquivos do programa são SQ111.COM e USQ120.COM, para compactar e descompactar, respectivamente. Os arquivos compactados no SQUEEZE ficam com um Q no meio da extensão. Por exemplo, COM vira CQM, LNK vira LQK e assim por diante.

GEL CRUNCHER

Este também veio do CP/M e é utilizado pelos usuários de modem há muito tempo. Não consegui descobrir o método utilizado por ele. O tempo de compactação e de descompactação é muito bom, informando sempre sobre o andamento da compactação. Os arquivos são CRUNCH.COM e UNCR.COM.

Os arquivos compactados pelo CRUNCH assumem um Z no meio da extensão, como um DZC, de DOC.

SHRINK

Este compactador foi obtido através de um fonte em C que compilei no MSX. É um compactador LZW puro, com um sistema bem simples de detecção de erros. Como foi feito em C, o tempo de compactação e de descompactação é absurdamente alto. A taxa de compressão no entanto ficou bem próxima à do GEL CRUNCHER. Os resultados dos testes deste programa foram omitidos, embora ele consiga uma taxa de compressão idêntica à do ARK (que segundo a própria documentação usa também o método LZW), os tempos de compactação são três vezes maiores que os do ARK. Acho que ele só vale mesmo para fins didáticos, por isso se alguém quiser os fontes, é só entrar em contato comigo através de CPU.

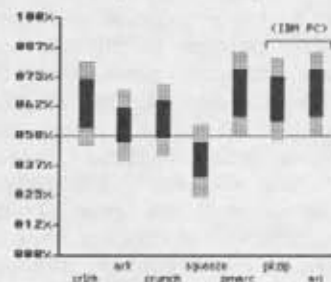
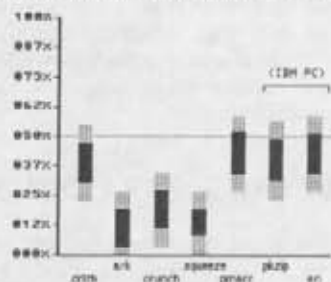
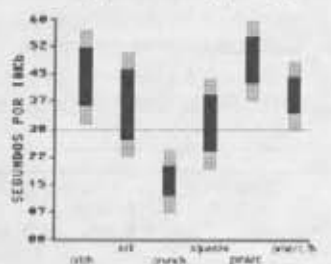
ARK/UNARC

Outro compactador de CP/M, compatível com o ARC do PC, que obteve uma taxa de compressão igual a do SHRINK. Segundo documentação do programa ele também utiliza o método LZW. A principal vantagem deste para os outros compactadores é que ele permite a compactação de vários arquivos em apenas um, sem necessidade de um "linker" para fazer isto. Na hora de descompactar, o usuário pode selecionar os arquivos desejados. Os arquivos são ARK.COM, que compacta arquivos gerando um ".ARK", e o UNARC.COM, que descompacta arquivos ".ARK" e ".ARC" (do PC). Uma observação quanto ao UNARC: A versão analisada por mim apresentou alguns problemas na descompressão de pacotes com muitos arquivos, talvez por alguma incompatibilidade entre o MSXDOS e o CP/M onde ele foi desenvolvido.

LZH CRUNCHER

Este compactador é bem recente. Foi desenvolvido para CP/M a partir de fontes em C que foram distribuídas pelos autores do programa original, Haruhiko Okumura e Haruyasu Yoshizaki. O método é chamado de LZHuf, que é o LZSS (??) do Okumura, modificado pelo Yoshizaki para também fazer o Huffman. Não obtive maiores informações sobre como seria este novo método, pois ainda não consegui esses fontes em C, mas imagino que o LZSS seria uma evolução do LZW, ou uma evolução da idéia dos dois primeiros autores do LZW, Lempel e Ziv.

Quanto à taxa de compactação, é muito melhor que a do GEL CRUNCHER e, segundo o autor do programa, poderia ser usado até para compactar arquivos já compactados no GEL CRUNCHER. Os resultados mostraram que ele consegue chegar bem próximo ao ZIP, com a vantagem de poder ser gerado no MSX. O ponto fraco dele é a velocidade, pois é muito mais lento que os outros programas. Os arquivos do programa são CRLZH.COM (compactador) UCRLZH.COM (descompactador) e um TYPELZH.COM (visualizador, opcional). Após compactar um arquivo no CRLZH ele fica com um "Y" no meio da extensão. Segundo informações que me foram passadas por quem trouxe este programa para o Brasil, já existe uma nova versão superior à essa versão 1.1. Vamos aguardar e ver como ficou o programa.

Uma característica importante do LZH CRUNCHER é que seu descompactador

Tempos de Compactação

Tempos de Descompactação

(UCRLZH) também descompacta arquivos gerados pelo GEL CRUNCHER e pelo SQUEEZE.

LHRD

A partir do mesmo fonte utilizado para criar o LZH CRUNCHER, foi elaborado, também para CP/M, o descompactador do LHARC do PC. Este descompactador, no entanto, está restrito a uma versão muito antiga do LHARC, sendo difícil encontrar arquivos que tenham sido compactados nesta versão (1.1).

Como eu mesmo não encontrei arquivos nesta versão, não fiz nenhum teste de tempo/taxa de compressão com este programa. Não é um programa muito útil, já que o PMEXT (visto adiante) pode descompactar arquivos do LHARC, mesmo da versão mais recente.

UNZIP 0.99

Esta versão estranha do UNZIP foi criada para CP/M e parece ser compatível com o PKZIP versão 1.1 do PC. Embora seja um programa lento, escrito em C, ele pode ser muito útil, pois o ZIP é um padrão muito popular no PC e sua versão 1.1 ainda é muito utilizada. Possui a mesma vantagem do ARK, ou seja, um arquivo ".ZIP", gerado no PC, pode ter vários arquivos dentro dele e estes podem ser descompactados separadamente.

Pelas informações que obtive, a criadora do ZIP do PC, a PKWARE, divulga os fontes apenas do UNZIP para estimular que sejam criados UNZIPs em outros micros. Por isso aparentemente não existe o ZIP para o MSX, mas segundo alguns contatos de CP/M nos EUA e no Canadá, o ZIP 2.04 já estaria pronto para CP/M. Esta versão, que foi lançada recentemente para PC, parece ser a que melhor comprime os arquivos.

Aparentemente o ZIP 1.1 é baseado no método Huffman, mas não obtive qualquer confirmação "oficial" a respeito disso.

Os arquivos que compõem o UNZIP 0.99 são UNZIP.COM, EXPLODE.OVR, UNSHRINK.OVR e UNREDUCE.OVR, sendo estes overlays chamados de acordo com o método utilizado na compactação.

UNARJ 2.30

Este programa eu mesmo adaptei para o MSX através alguns fontes. Por ser um programa totalmente escrito em C, é muito lento e infelizmente só roda em micros MSX2 (ou superior) ou em um MSX1 com Megaram (estou preparando uma versão para Memory Mapper). Isso porque, por ser um padrão bem sofisticado, baseado no método Huffman e com algumas melhorias, precisa de mais de 64 Kb de RAM. A taxa de compressão do ARJ 2.30 é melhor que a do ZIP e do LZH.

O ARJ também pode colocar vários arquivos em um só e o UNARJ do MSX pode selecionar os arquivos que devem ser descompactados. O programa é composto de apenas um arquivo, o UNARJ.COM. Ah! Só para justificar os tempos obtidos nos testes, como não terminei a versão para Megaram, tive que testar com a versão para VRAM, onde o acesso é mais lento do que na Megaram. Aposto que a versão para Megaram será mais rápida do que o Unzip!

PMARC/PMEXT

Este compactador/descompactador, que consegui pouco antes de fechar o artigo, foi realmente a grande revelação. Eu, que já utilizo compactadores há um bom tempo, ao testar este programa, achei fantástico. De autoria do japonês Yoshihiko Mino, este programa obteve um taxa de compressão praticamente idêntica a do ARJ 2.30. Pelo que pude observar, o programa parecer utilizar o mesmo algoritmo, talvez melhorado, do LHARC versão 2.13 do PC. Este programa pode empacotar vários arquivos em um só, como o ARC, gerando uma extensão PMA.

Já o PMEXT (descompactador) pode descompactar ".PMA", ".LZH" (do PC) e ".COM". Talvez o leitor tenha estranhado o ".COM", mas aí está uma das novidades deste programa: ele pode montar arquivos "self-extract", ou seja, autodescompactáveis – um arquivo que pode se descompactar, criando vários outros! Assim você não precisa usar sempre o PMEXT, podendo levar um programa para alguém de forma compactada sem precisar levar o descompactador! Além disso, tanto o PMEXT quanto o arquivo ".COM" (self-extract) podem executar um arquivo compactado sem precisar criá-lo em disco.

O único inconveniente do PMARC/PMEXT é que, por ser um compactador bem complexo, a descompressão e execução de um programa se torna um processo muito lento.

O PMARC possui dois modos de operação: o normal e o de alta velocidade (opção /H). No modo de alta velocidade, embora obtenha-se uma taxa de compressão um pouquinho menor, fica cerca de 15% mais veloz. Veja a tabela de comparação de velocidade dos compactadores. No pacote do programa vem ainda um "concatenador" de arquivos compactados, e alguns arquivos de documentação... em japonês (alguém quer traduzir?).

ARQUIVADORES

Para poder compactar vários arquivos em um só, em programas como o LZH CRUNCHER, foram criados alguns "arquivadores". Os mais utilizados são o LINKER, que trabalha com um formato próprio, desenvolvido por Darci Marcondes Filho, e gera um arquivo ".LNK" com os arquivos escolhidos e o LBR, um padrão muito utilizado no CP/M.

Para montar um LBR existem vários programas, como o NULU. O arquivo ".LNK" pode ainda ser compactado, mas, se for preciso retirar apenas um arquivo do LNK, será necessário descompactar o arquivo inteiro e retirar todos os arquivos do pacote. Já no LBR, os arquivos devem entrar no LBR já compactados. Dessa forma, é possível selecionar apenas um para descompactação (nada impede o usuário de compactar um LBR já montado, mas isto não é comum).

Além destes programas específicos para alguns padrões, existem alguns programas como o LT29.COM que descompactam em vários métodos. O LT, pode descompactar arquivos gerados no SQUEEZE, no GEL CRUNCHER e no LZH CRUNCHER, dentro ou fora de LBRs.

CONCLUSÃO

Levando em conta os resultados de tempo/desempenho obtidos nos testes e a operacionalidade dos programas, podemos dividi-los segundo sua adequação.

O PMARC pareceu realmente ser o melhor. Embora não seja um programa muito rápido como o CRUNCHER, ele tem uma taxa de compressão excelente, possui recursos de arquivar vários arquivos em apenas um, criar "self-extracts" e dois modos de operação. Como programa mais completo para o MSX ele merece uma atenção muito especial.

O GEL CRUNCHER apresentou-se como um programa muito rápido e com uma boa eficiência. Muito bom quando necessita-se compactar algo sem perder tempo.

O Master Cruncher, embora apresente resultados baixos em taxa de compressão, é recomendado para telas, que não podem ocupar muito espaço e ao mesmo tempo devem poder ser visualizadas a qualquer momento. A vantagem dele em relação ao PMARC é que o arquivo pode ser rapidamente descompactado.

O UNZIP, o UNARJ e também o PMEXT podem ser muito úteis para trocar dados entre o PC e o MSX, com uma taxa de compressão muito alta. ❑

Miguel Freitas tem 14 anos e cursa a 8ª série do 1º grau. É autor do programa de comunicação MSXANSI e pode ser encontrado pelas BBSs da Fidonet no Rio de Janeiro.

LISTAGEM 1

```
10 ' Programa exemplo para demonstrar a
 utilização
20 ' do método Run-Length em arquivos t
exto.
30 ' Para a revista CPU, por Miguel Fre
itas
40 '
50 ' Para que haja uma boa redução no t
amanho, o arquivo
60 ' deve conter repetições de bytes, c
omo por exemplo:
70 ' "- - - - - - - - - - - - - - - -".
80 '
90 MAXFILES=2:CLS
100 PRINT "Run-Length para arquivos -te
xto":PRINT
110 PRINT "1 - Compactar arquivo"
120 PRINT "2 - Descompactar arquivo"
130 PRINT "3 - Ver arquivo compactado"
140 PRINT "4 - Basic":PRINT
150 PRINT "Escolha:";:A$=INPUT$(1)
160 IF A$<"1" OR A$>"4" THEN RUN
170 IF A$="4" THEN END
180 IF A$="2" THEN 310
190 IF A$="3" THEN PRINT:LINE INPUT "No
me do arquivo:";NA$:NB$="crt:":GOTO 330
200 PRINT:LINE INPUT "Nome do arquivo f
onte:";NA$
210 LINE INPUT"Nome do arquivo destino:
";NB$
220 OPEN NA$ FOR INPUT AS #1
230 OPEN NB$ FOR OUTPUT AS #2:CNT = 0:S
$ = "":TE = 0:TS = 0
240 IF EOF(1) THEN IF CNT = 1 THEN 300:
ELSE GOTO 260
250 TE = TE + 1:A$=INPUT$(1,#1):IF A$ =
 S$ THEN CNT = CNT + 1:GOTO 240
260 IF CNT>2 THEN PRINT #2,CHR$(255)CHR
$(CNT)S$;:TS = TS + 3:CNT = 1:S$ = A$:G
OTO 240:' máximo de 255 repetições (1
byte)
270 IF CNT=1 THEN PRINT #2,S$;:TS = TS
+ 1
280 IF CNT=2 THEN PRINT #2,S$S$;:TS = T
S + 2
290 S$ = A$:CNT = 1:GOTO 240
300 PRINT #2,S$;:CLOSE:PRINT "Arquivo o
riginal:"TE:PRINT "Arquivo compactado:"
TS:PRINT "Total economizado:"TE-TS:A$=I
NPUT$(1):RUN
310 PRINT:LINE INPUT "Nome do arquivo f
onte:";NA$
320 LINE INPUT "Nome do arquivo destino
:";NB$
330 OPEN NA$ FOR INPUT AS #1
340 OPEN NB$ FOR OUTPUT AS #2
350 IF EOF(1) THEN PRINT:CLOSE:PRINT "O
K!":A$=INPUT$(1):RUN
360 A$=INPUT$(1,#1):IF A$ = CHR$(255) T
HEN 380
370 PRINT #2,A$;:GOTO 350
380 G=ASC(INPUT$(1,#1)):A$=INPUT$(1,#1)
390 FOR F = 1 TO G:PRINT #2,A$;:NEXT:GO
TO 350
```

MSX BITS

Interface alternativa de I/O

Adriano C. R. da Cunha
Fernando Blanco

Acredito que, assim como eu, muitos leitores de CPU não conseguiram montar satisfatoriamente o Projeto Hardware. Para contornar o problema, apresento aqui uma modesta interface de I/O para seu MSX. Possui oito canais de saída e seis de entrada (expansível para vinte canais).

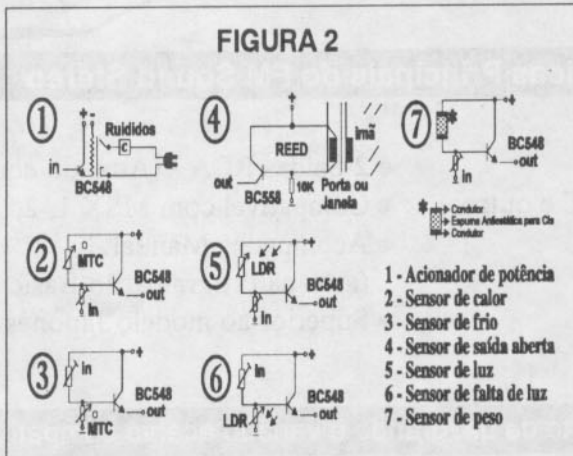
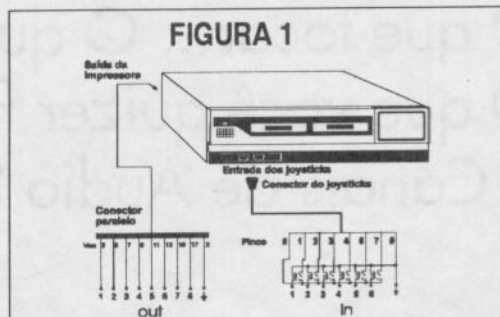
Os canais de entrada utilizados são os do joystick e para saída utilizamos a interface da impressora. Na figura 1 está o esquema do projeto.

A leitura das portas é feita com a função STICK e STRIG do Basic ou 00D5H e 00D8H do BIOS. O envio de sinais é feito com um número binário de oito dígitos, sendo 1 para saída ligada e 0 para desligada, via porta 91H. O programa da listagem 1 ilustra este procedimento.

Na figura 2 temos diversos circuitos de sensores para serem utilizados pela interface. Caso o circuito utilizado não necessite de alta corrente, pode-se usar a fonte do micro, caso contrário será necessário o uso de uma fonte externa.

Quase todos os circuitos do Projeto Hardware funcionarão (apenas os que usem menos de 7 ou 9 portas), bastando para isto algumas modificações apenas. ❑

FIGURA 1

FIGURA 2

LISTAGEM 1

```
10 CLS:PRINT"Interface
alternativa":PRINT:A$="00000000":POKE
&HFCAB,255
20 INPUT"Ler ou Escrever";S$:IF S$="L"
THEN GOTO 60
30 INPUT"Porta (1-8)";P:INPUT"Seta ou
Reseta";S$
40 MID$(A$,9-P,1)=CHR$(48-(S$="S")):
A=VAL("&B"+A$)
50 OUT &H91,A:GOTO 10
60 INPUT"Porta (1-6)";P:P=P+P-1:IF P<5
THEN P=STICK(P)
70 IF P=5 THEN P=STRIG(1) ELSE IF P=6
THEN P=STRIG(2)
80 A$(0)="setada":A$(1)="resetada":PRI
NT"Porta ";
90 PRINT A$(-(P=0)):PRINT:GOTO 10
```

FM SOUND STEREO

Este bichinho vai dar o que falar... Ou melhor dizendo! O que tocar... O que bater... O que você quizer fazer com 9 Canais de Audio Stereo.

Características Principais do FM Sound Stereo

- 9 canais de áudio stereo (FM)
- Compatível com MSX-Music (FM PAC e outros)
- Soma-se aos 3 canais do PSG (totalizando, 12 canais stereo)
- O PSG possui chave on-off
- 2 Saídas RCA - (Aux. do equipamento de som)
- Compatível com MSX 1, 2, 2+ e Turbo R
- Acompanha Manual (aux. na Programação Basic e Assembly)
- Superior ao modelo Japonês em qualidade sonora.

A relação de softwares abaixo fazem parte de um grande e variado acervo compatível com FM Sound Stereo

JOGOS

Akambe dragon, Aleste 1-2-3, Animal`s Wars 2, Arcus 2 e 3, Burai, Columbus, Dragon Slayer 6, Dragon Quiz, Emerald Dragon, F1 Spirit 3D Special, F1 Time Pilot, Famile Parodic 2, Feed-back, Fray, Gouvellius 2, Great Test Driver, Hy-defos, Illusion City, Jump Hero 2, Kaguerou Meikyu, Laydock 2+, Mid Garts, Magnar, Mon Mon Monster, Nyan, Nyancle Racing, Pacmania, Palacio, Phantasie 3, Pipe line Degorby, Pink Sox 1 a 8, Pyou Pyou, Playball 3, Princess Maker, Psycho World, Psi-o-Blade, Quinpl, Randar 2 e 3, Rune Worth, Rune Master 1 e 2, Sea Sardine, Seed of Dragon, Sorcerian, Super Cooks, Super Zelister, Suchaughauan, Tetris 2, Thexder 2, Twinkle Star, Undeadline, Usa Jong, Valis 2, Xak 1, 2 e 3, Xevious.

APLICATIVOS & Cia.

Beppin, Bcf Disk Station 1 a 8, Cheat Disk, Club Guide Disc 1 a 6, Club Picture 1 a 10, Crackbird FM Demo, Disk Special T&SOFT 1 a 6,Dante Constructor 1 e 2, Disk Album 34, Disk Station Special 1 a 8, Disk Station 1 a 35, Disk PAC 1 a 3, Disk Pac Elfo Soft, Disk Fan 1 a 36, Demo do Sony HB-F1XDJ1, Demo Dragon Disk 1 a 12, Fac Demo 2, Fac Soundtracker 1 e 2, FM Basic Collection 1 a 6, FM T&Soft Collection, FM POP Collection,Fm Fac demo, Future Magazine 1 a 6, Lighting Demo, Mgsel Driver, Myadock, Opll Driv-er, Peach Up 1 a 8, Synth Saurus, Saurus Lunch 1 a 8, Studio Fm Promo, Sum Pac 1 e 2, Synth Power 2, The Ultimate Rax Demo, Turbo Sma

Um Produto Exclusivo

TecnoBytes

Informações Técnicas c/ Tecnobytes
Cx Postal 79841- C. Rocha
S. João de Meriti - CEP:25550-970-RJ

Informações Gerais e Vendas
Takeru Software
Tel:(021) 231-2335

Criação: Wallace Henrique

MSX View

O 'Windows' do MSX

Edison Antonio Pires de Moraes

Seguindo a tendência mundial de interfaces gráficas, o padrão MSX conta atualmente com vários sistemas operacionais gráficos, como o Easy Working, o SuperDOS e o EasyDOS para MSX2 carregado com o MSXDOS 2.2. Entretanto, o ambiente operacional gráfico padrão dos micros MSX, criado pela ASCII/Microsoft, é o MSX View que roda somente nos modelos MSX Turbo R sob o MSXDOS 2.3. O MSXView é dividido em um módulo principal, o VSHELL e em quatro módulos básicos: ViewDRAW (Desktop Publishing), ViewTED (processador de textos), ViewPAINT (editor gráfico bit-mapped) e o PageBOOK System (gerenciador para desktop publishing) que se divide em três submódulos: PageEDIT, PageLINK e PageVIEW.

VISUAL SHELL

Ao carregar o MSXView, o primeiro módulo que aparece é o VSHELL (Visual SHELL). Através do VSHELL pode-se fazer qualquer operação do MSXDOS 2 com muito mais facilidade, através de menus "pull-down", bastando colocar o apontador sobre a opção desejada e teclar o botão esquerdo do mouse ou GRAPH+SELECT no teclado. Qualquer opção pode ser cancelada com o botão direito do mouse ou GRAPH+STOP no teclado. Para mover o apontador pelo teclado, pode-se usar GRAPH+SETAS. Observe a figura 1 para entender melhor o VSHELL.

1 - Ícones correspondentes aos arquivos do diretório ou subdiretório atual. Esses ícones podem ser trocados de lugar, editados etc.

2 - Drive atual. No total, podem haver até oito drives, de A: até H:, sendo que o drive H: é reservado pelo sistema como RamDisk.

3 - Nome do subdiretório atual.

4 - Versão do VSHELL. Ao colocar o apontador sobre essa opção e selecioná-la, aparecerá o seguinte menu:

DRIVE SELECT
LOAD
FILE INFO
DISK INFO
FORMAT DISK
PRINT FORMAT
PRINT
QUIT

Através desse menu, pode-se selecionar o drive, obter informações sobre o disco ou sobre um arquivo previamente selecionado, formatar o disco, selecionar o tamanho do papel para a impressora (normalmente o tamanho é o A4) e outros. Pode-se também sair do VSHELL e voltar ao MSXDOS2 através da opção QUIT.

5 - View. Ao selecionar essa opção, aparecerá um menu com os arquivos que são executáveis sob o VSHELL (da mesma forma que os arquivos ".COM" são executáveis sob o MSXDOS2). Alguns arquivos já vêm com o MSXView, valendo destacar:

SYSTEM - Seleciona mouse ou teclado, inclusive com graduação de "sensibilidade" de deslizamento do mouse.

PRINTER - Seleciona uma entre 15 impressoras de padrões diferentes. Destas, 14 são impressoras de 24 agulhas. A impressora "default" é a FS-PC1 da Panasonic, uma impressora colorida de 48 agulhas e com resolução de 400 dpi, semelhante a de impressoras laser. Entretanto, outros filtros de impressoras podem ser adicionados para adaptar o MSXView às impressoras de 9 agulhas comuns no

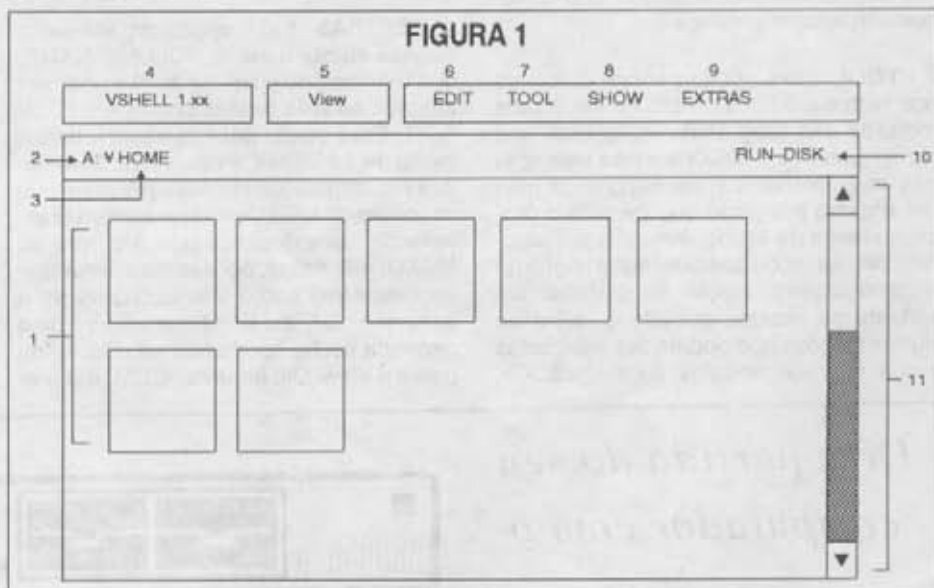



FIGURA 1

Brasil, embora com grande perde de qualidade do resultado final.
SCREEN - Muda as cores da tela (inicialmente de fundo branco com caracteres pretos) e faz o ajuste de tela, como o comando SET ADJUST do BASIC.

Existem vários outros arquivos executáveis, como o CLOCK que exibe um relógio de ponteiros na tela, o CALENDAR, que exibe um calendário, o MEMO, que é uma pequena agenda eletrônica, além de vários outros.

6 - EDIT. Essa opção é normalmente usada após a seleção de um arquivo. Ao ser selecionada, aparecerá o seguinte menu:

MAKE DIRECTORY
RENAME
DELETE
COPY
MOVE
SELECT ALL
CANCEL

Como se pode ver, é aqui que está a maioria das funções de arquivos realizadas pelo MSXDOS2. Pode-se copiar, renomear, apagar arquivos etc.

7 - TOOL. Essa opção permite o acesso aos outros módulos do MSXView. Esses módulos são programas integrados que fazem com que o MSXView não seja apenas um ambiente operacional gráfico, mas um sistema integrado que permite o processamento de textos, inclusive em japonês, português ou qualquer outra língua no mesmo arquivo, edição de gráficos, um sistema de desktop publishing, além de outras funções que podem ser integradas pelos três submódulos do PageBOOK System. As ferramentas que estão disponíveis nessa opção são as seguintes: DRAW, TED, PAINT, PAGEEDIT, PAGELINK e PAGEVIEW. Esses três últimos pertencentes ao PageBOOK System.
8 - SHOW. Ao selecionar essa opção, aparece o seguinte menu:

ICON
NAME TYPE
SIZE DATE
ORGANIZE
STANDARD

Este é um menu para a seleção do formato de apresentação do diretório pelo VSHELL. Ao selecionar ICON, o diretório é apresentado através de ícones. As outras quatro opções (NAME, TYPE, SIZE e DATE) servem, respectivamente, para apresentar o diretório organizado por ordem de nome, tipo, tamanho ou data. Nesses quatro casos, os arquivos são mostrados no mesmo formato do MSXDOS2 e não por ícones. Para ver o diretório da forma como está no disquete, use a opção STANDARD. A opção ORGANIZE reorganiza os arquivos.

9 - EXTRAS. Essa opção dá acesso a apenas outras duas, o VOLUME NAME, que é o nome do disquete atual e pode ser alterado através dessas opção, e a ICON EDIT. Essa opção permite editar o banco inicial de 19 ícones. Pode-se também deletá-los, ampliá-los etc. Vale ressaltar que os ícones do MSXView representam graficamente as extensões dos arquivos do MSXDOS2. Assim, por exemplo, um arquivo executável sob o VSHELL deve ter a extensão ".DA" e o ícone respectivo é uma pequena flecha apontando na direção da palavra View. Um arquivo ".COM" é representado pelo desenho de um microcomputador com os caracteres "A" na tela. Se uma extensão não é reconhecida pelo VSHELL, aparece um ponto de interrogação. Enfim, os ícones são reflexo direto das extensões dos arquivos do MSXDOS2.
10 - Esse é o nome do disco corrente. O nome pode ser modificado pela opção anterior (VOLUME NAME) ou através do MSXDOS2 pelo comando VOL.
11 - Quando o diretório é muito grande e não cabe inteiro na tela, pode-se usar essa opção para fazer um scroll vertical do diretório, para que todos os arquivos possam ser vistos.

Esse é um resumo das funções do VSHELL. Vale ressaltar que quando carregado na memória, o VSHELL fica inteiramente residente, evitando que o disco de sistema seja solicitado a toda hora. Aliás, todos os módulos do MSXView ficam inteiramente residentes na memória.

ViewDRAW

Até agora, falamos apenas do VSHELL, que é o módulo básico do MSXView. Vamos falar um pouco do ViewDRAW.

O ViewDRAW é um editor para Desktop Publishing com gráficos vetoriais. Com ele pode-se editar uma tela que depois pode ser impressa em papel, normalmente no tamanho oficial A4, embora outros tamanhos estejam disponíveis. Como a folha toda não cabe na tela, pode-se fazer um scroll vertical ou horizontal para se ter acesso a toda a página editada. Até 10 páginas podem ser editadas simultaneamente.

Toda a edição pelo ViewDRAW é feita através de vetores. A edição vetorial pos-

sui inúmeras vantagens em relação à edição direta na tela, com um gasto mínimo de memória e com uma incrível facilidade para a correção de erros. Mas o que é edição vetorial? Podemos dizer que é uma edição através de "janelas". Para se escrever um texto, por exemplo, abre-se uma janela no vídeo de tamanho suficiente para conter esse texto. Dentro dessa janela, o texto pode ser editado como em um processador de textos comum. Porém, cada janela só pode conter um tipo de letra. Para escrever um título para esse texto, com letras maiores, deve-se abrir outra janela. Essas "janelas" podem ser movidas à vontade por toda a página até estabelecer-se o lugar mais adequado.

Além disso, o ViewDRAW também permite a edição de linhas, retângulos, círculos, elipses etc., que podem ser preenchidos com vários tipos de texturas e movidos livremente por toda a página, como as janelas de texto.

Inicialmente, conta-se com um banco de apenas 11 tipos diferente de letras, mas que podem ser ampliados indefinidamente. Conta-se também com sete tipos de formatos (itálico, vazado, sombreado etc.) que podem ser misturados uns com os outros e com 10 tamanhos diferentes de caracteres, independente do tamanho original, desde 8x8 pontos até 32x32 pontos. Além disso, o texto pode ser ajustado à esquerda, direita ou no centro da janela respectiva.

Uma característica interessante do editor de textos do ViewDRAW é que as letras não ocupam sempre o mesmo espaço na janela. Num editor comum, um "j" ocupa a mesma largura que um "m", por exemplo oito pontos horizontais. No ViewDRAW, um "j" ocupa bem menos espaço que um "m", proporcionando um espaço mais uniforme entre as letras e um resultado final bem mais elegante.

ViewPAINT

Outro módulo do MSXView é o ViewPAINT, um editor gráfico "bit-mapped", no estilo tradicional. Não há muito o que comentar sobre esse módulo. Ele tem tudo o que todo bom editor gráfico tem. Pode-se editar linhas, pontos, círculos, elipses, retângulos e a partir desses recursos, criar figuras mais complexas, inclusive com a ajuda do "zoom". Há também a possibilidade de escrever algum texto, preencher uma área com inúmeros tipos de texturas diferentes etc. Enfim, o VIewPAINT é um poderoso editor gráfico com inúmeros recursos e que pode ser integrado aos outros módulos do MSXView através do PageBOOK System.

ViewTED

Mais um módulo do sistema, o ViewTED é um editor de textos com modernos recursos de edição. Um detalhe que o difere de outros editores de texto é que ele usa o set de caracteres Kanji JIS1 e JIS2 do micro, possibilitando o acesso à cerca de seis mil Kanjis (que fazem parte do alfabeto japonês), aos alfabetos grego e russo completos e às centenas de caracteres especiais, podendo assim editar textos em japonês, chinês, grego, russo, inglês etc. Nem mesmo o nosso amado português foi esquecido, já que há caracteres acentuados disponíveis. Trata-se de um processador de texto com recursos modernos e poderosos.

PageBOOK

Os três módulos seguintes, o PageEDIT, PageLINK e PageVIEW fazem parte do PageBOOK System. Esses módulos servem para integrar os outros módulos do MSXView. Por exemplo, o ViewDRAW, apesar se ser um poderoso editor de Desktop Publishing, não aceita os tradicionais shapes ou figuras para edição. Assim, através do PageBOOK System podemos inserir figuras editadas pelo ViewPAINT no ViewDRAW. Esse é apenas um dos inúmeros recursos que o PageBOOK System oferece. Cada um dos três submódulos permite operações distintas, desde a edição até a montagem final de todos os módulos integrados.

CONCLUSÃO

O MSXView não é apenas um ambiente operacional gráfico, mas um verdadeiro sistema integrado com poderosíssimos recursos o que dispensa a aquisição de muitos softs que normalmente seriam necessários para se trabalhar com um microcomputador.

Para finalizar, vale aqui ressaltar que o segundo modelo MSX Turbo R, o FS-A1GT da Panasonic já vem com o MSXView residente, na forma de uma "ROM-DISK", designada por "C:", onde estão gravados todos os 35 arquivos que compõem o MSXView. Existe também um quarto "drive" interno, designado por "D:" e chamado de SRAM-DISK, que é uma RAM mantida a bateria, de apenas 16 Kbytes, e que serve unicamente para guardar a configuração atual do MSXView, já que a ROM não pode ser gravada. Apesar do desperdício de drives, ainda assim sobram três para serem conectados (E:, F: e G:) para acoplar-se, por exemplo, mais dois drives externos e um Winchester. O oitavo drive, designado por "H:" é reservado pelo MSXDOS2 como RAMDISK. Além destes, evidentemente, existem ainda os drives "A:" e "B:".

Já para o primeiro modelo MSX turbo R, o FS-A1ST, o MSXView deve ser obtido em disquete. Dos três disquetes que compõem o sistema, o primeiro deve ser mantido desprotegido contra gravação, pois o sistema faz constantes atualizações no mesmo, da mesma faz na SRAM-DISK do FS-A1GT.

Na Europa já existe uma versão em inglês do MSXView, mas, ao que eu saiba, esta ainda não chegou ao Brasil. Apenas a versão em japonês apareceu por aqui. É claro que a versão do A1GT é em japonês. Mas a versão em inglês não deve demorar muito para chegar por aqui. ❑

MSX ANSI

O máximo em comunicações

Sempre que um bom software é lançado para o MSX, podemos constatar que este micro ainda tem muito a oferecer e os produtores de soft estão aí, com a corda toda. Uma boa prova disso é o lançamento de um dos mais completos softs de comunicação para a linha MSX, o MSX ANSI de Miguel Freitas.

O MSX ANSI foi feito visando praticidade e conforto ao usuário, mas nem por isso deixa de ser um excelente e completo soft de comunicação.

CARACTERÍSTICAS GERAIS

Dispondo de menus apresentados através de janelas e com funções acessadas de forma objetiva e direta por *Hot-keys*, o MSX ANSI possui também *Macro-keys*, frases prontas de até 80 caracteres.

As *Macro-keys* são muito úteis ao acessar uma BBS, onde, em vez de digitar seu nome todo, senha etc, entra-se com uma macro pré-definida. Ou ainda, basta comandar uma *Macro-key* com seus dados para cadastrar-se *On-line* em uma nova BBS, tornando-se desnecessário digitar todos os seus dados manualmente.

Outros dois recursos muito úteis e interessantes são a discagem e a resposta automática, embora alguns modems não possam ser beneficiados por estes recursos, por questões de hardware.

A versão do MSX ANSI aqui analisada é compatível com os modens TEL-COM e DDX, mas a versão do programa para o TM-2 da Gradiente estará disponível em breve.

CONEXÕES

O MSX ANSI faz uso de todas as velocidades apresentadas em seu menu de seleção, com exceção da velocidade de 75/1200 BPS (Bits Por Segundo), que, assim como a discagem e a resposta automática, não está disponível em alguns modems.

Na velocidade de 75/1200, o usuário recebe a 75 BPS (lento) e transmite a 1200 BPS (rápido). Já na velocidade de 1200/75 ocorre exatamente o oposto, ou seja, o usuário recebe a 1200 BPS e transmite a 75 BPS.

Nas velocidades de 300 BPS existem dois padrões, 300 Bell (padrão americano) e 300 CCITT (padrão europeu), que também estão disponíveis. Pode-se optar pelo modo "origem" ou "resposta", que geram tons diferentes, possibilitando que os micros conectados recebam e transmitam dados simultaneamente.

A conexão pode ser feita de duas formas: diretamente, através de uma opção do menu principal, ou através da discagem automática, onde o usuário pode discar para mais de um número. Para isto o MSX ANSI conta com um banco de 20 números de BBS definíveis pelo usuário. Assim, se o primeiro número estiver ocupado, o modem disca para o segundo e assim por diante, até que se consiga uma conexão.

O MSX ANSI é dotado do protocolo XMODEM, o que possibilita a transferência de arquivos *On-line*. Trocando em miúdos, isto permite a troca de programas pela linha telefônica. O protocolo XMODEM usa o padrão CRC (Cyclical Redundancy Check) para a verificação dos dados. Essa checagem é feita para que se possa certificar que o arquivo transferido não sofreu danos por "impurezas" na linha telefônica. Se algum problema for detectado, o XMODEM corrige o erro. Existem protocolos mais recentes e complexos que o XMODEM, porém este é o mais tradicional e o mais usado no padrão MSX.

Ao contrário dos protocolos mais modernos, o XMODEM requer o nome do arquivo que será enviado ou recebido, o que não ocorre naqueles protocolos, que informam este nome ao outro micro. Para facilitar esta operação, o MSX ANSI procura algum nome de arquivo válido que esteja na tela e o assume como default.

OUTRAS CARACTERÍSTICAS

O soft é muito versátil, permitindo que se use tanto a tabela original de caracteres MSX, como a tabela padrão PC. A tabela do PC é útil para visualizar as telas como elas são constituídas originalmente nas BBS.

Está disponível ainda o recurso do relógio, que funciona como um "timer", onde pode-se selecionar uma determinada hora para o MSX ANSI iniciar a discagem automática, mas este recurso só funciona em máquinas que possuem relógio real.

É possível armazenar em um *buffer* as últimas 48 linhas recebidas, que podem ser visualizadas a qualquer momento durante a conexão. Outra manei-

ra de armazenar o texto é utilizando uma opção para gravar em disco tudo o que vier para tela. Além disso, pode-se enviar um texto pronto, no formato ASCII. Se o usuário possuir uma impressora, pode copiar nela tudo que for recebido na tela.

Uma opção muito útil é a apresentação do diretório. Quando solicitada, esta opção exibe o nome do arquivo, tamanho, número de blocos, tempo de transferência (em 300 e em 1200 BPS) e o espaço livre no disco.

O modo HOST, um outro recurso do MSX ANSI, pode ser ativado diretamente, ou ainda habilitar a resposta automática. Desta forma, o usuário pode até sair de casa e deixar o micro ligado em *stand by*, caso esteja esperando por um *download* ou por um *upload* (transmissão ou recepção de um arquivo respectivamente).

O vídeo pode ser configurado de duas formas: com 40 colunas por 24 linhas ou 80 colunas por 25 linhas (só disponível para máquinas MSX 2 em diante). Na parte inferior do vídeo existe a linha de status onde é exibida a configuração atual do MSX ANSI. Teclando-se "Shift", esta linha mostra outras informações.

No modo CHAT, o MSX ANSI divide o vídeo em duas partes. Na parte superior, só aparece o que for recebido e na parte inferior só o que for enviado, permitindo que os usuários dos micros conectados enviem e recebam caracteres simultaneamente, sem haver "aquela" embolação na tela que ocorre nos softs tradicionais.

Outra característica interessante do MSX ANSI é a de tocar músicas para avisar ao SysOp (System Operator), quando lhe for pedido um CHAT a partir do menu HOST, que houve uma conexão bem-sucedida através da discagem automática ou ainda para alertar sobre o fim da transferência de um arquivo.

A tecla "Select" é de grande importância, pois com ela pode-se voltar para o menu principal em qualquer ponto do soft, mesmo durante uma comunicação, permitindo que se façam as alterações necessárias, sem que a linha caia.

O ANSI, ENFIM

O ANSI é um padrão do American National Standards Institute que define vários códigos de controle para o tratamento do vídeo. É muito útil nas BBS, por exemplo, em situações nas quais o usuário precisa fazer uma alteração na tela sem ter que reescrevê-la totalmente. Para isto é só ativar o ANSI para que seja feita a alteração local, sem que a tela tenha que ser toda refeita.

Entretanto, na velocidade de 300 BPS, é recomendável não usar o ANSI todo o tempo, pois isso tornará a comunicação muito lenta. O ANSI deve ser ativado no "TimeBank", nos jogos *Online* e nos demais serviços que oferecem este padrão, já que o ANSI é de grande importância e utilidade para melhorar a operacionalidade dos softwares que o utilizam.

Acompanham o MSX ANSI um manual digital detalhado com chamadas em hipertexto, um programa que calcula os pulsos e os minutos das chamadas que o usuário faz com o MSX ANSI, além do custo da ligação, segundo o padrão Telebrás de cobrança, e outro que faz estatísticas do uso do modem.

Resumindo: O MSX ANSI é simplesmente fantástico! É ver para crer! ❑

Análise e texto final: Vinicius Beltrão.

Memory Mapper

Um super projeto em duas partes

R. Pontes
Roberto Silva

Muitas vezes o usuário compra jogos ou aplicativos, mas após várias tentativas malsucedidas de rodá-los, acabam achando que seu micro está com defeito. Entretanto, após outros testes, constatam que seu equipamento está funcionando perfeitamente. Surge então a indignação e a revolta, pois ao voltar à softhouse para reclamar, recebe uma resposta irritante: "Seu micro não tem Memory Mapper e sem essa expansão os programas nunca irão rodar". Com isso o usuário se questiona se vale a pena comprar mais uma expansão de memória, outra expansão semelhante a Megaram, velha conhecida dos usuários do MSX.

A verdade, porém, é que essa "nova" expansão é a verdadeira expansão de memória para MSX. A seguir vamos enumerar algumas vantagens de possuir uma Memory Mapper interna:

- A Memory Mapper segue à risca o padrão MSX, sendo reconhecida pelo sistema como expansão real de memória, o que não ocorre com a Megaram.
- Não ocupa os slots externos, já tão escassos nos nossos micros.
- É utilizada por 95% dos programas europeus, que são inúmeros e não apenas uma meia dúzia, como é o caso da Megaram. Dentre eles podemos destacar os seguintes:

    Dynamic Publisher
    Ease
    Final Graphics
    Video Graphic Philips
    The Animator

- Todos os jogos de Megarom já estão adaptados para Mapper. Boa parte destes jogos, além de aplicativos, demos e utilitários já se encontram disponíveis no Brasil.
- Muitos jogos de Megarom (de onde se originou a Megaram) utilizam interfaces de som diferentes do PSG (chip de som interno do MSX) como é o caso da Konami, que usa o SCC, e de outros fabricantes que utilizam o FM. Mas em se tratando de Megaram fica impossível utilizar tais interfaces já que os dois slots externos do micro ficam ocupados (um com a Megaram e outro com a interface de drive). Já com a Mapper interna tal inconveniente não ocorre, pois ela nos deixa um slot livre no qual podemos conectar um cartucho de SCC ou de FM, permitindo assim ouvir a trilha sonora original.

- A Mapper pode ser manipulada em Basic, coisa que não e possível com a Megaram.
- Seu acesso e rápido, fácil e eficiente como já foi descrito em artigos anteriores.
- Já que a Memory Mapper descrita neste artigo é de instalação interna, boa parte do circuito necessário à montagem da Mapper já está contida no próprio micro. Com isto, reduz-se drasticamente o número de componentes e, conseqüentemente, o seu custo final.

DESCRIÇÃO DO PROJETO

Nosso projeto é uma Memory Mapper de 256 Kbytes, de instalação interna, para micros MSX 2 e MSX 2+, que irá ocupar o slot do banco de RAM de 64 Kbytes, substituindo-o por outro de 256 Kbytes. O usuário pode estar se perguntando porque apenas no MSX 2 e MSX 2+? A resposta é simples: o BIOS (ROM) destes micros inicializa os registros da Memory Mapper sem os quais com certeza ocorreria um belo "crash" no sistema. O BIOS do MSX 2+ vai ainda mais longe, nos mostrando em sua inicialização o tamanho da Mapper com a mensagem "Main RAM: tamanho". No nosso caso aparecerá: "Main RAM: 256Kbytes".

A Memory Mapper divide o banco de 256 Kbytes em 16 blocos de 16 Kbytes cada e para gerenciar estes blocos a Mapper dispõe de 4 registros (R0 a R3) de 4 bits cada. Estes registros podem conter um número de 0 a 15 que correspondem ao número de blocos do banco de 256 Kb. Os registros são acessados pela CPU através de portas nas quais podemos ler ou escrever:

Registro	Porta de acesso
R0	FCH
R1	FDH
R2	FEH
R3	FFH

Cada um dos quatro registros é responsável pelo gerenciamento de uma página de memória, a saber:

Registro:	Página
R0	Pág. 0
R1	Pág. 1
R2	Pág. 2
R3	Pág. 3

Para que um determinado registro da Mapper aponte para um determinado bloco do banco de 256 Kb basta escrever na porta de acesso do respectivo registro o número do bloco a ser "mapeado".

Por exemplo, vamos supor que o usuário queira que a página 01 de memória seja o bloco número 08 do banco de 256 Kb. O responsável pelo "mapeamento" da página 01 é o registro R1 da Mapper e sua porta de acesso é FDH. Portanto basta que o usuário escreva nesta porta o número do bloco, ou seja:

```
OUT &HFD,08
```

Em um MSX sem a Mapper interna as linhas de endereço A15 e A14 fazem a seleção das páginas de memória:

A15	A14	Página
0	0	Pág. 0
0	1	Pág. 1
1	0	Pág. 2
1	1	Pág. 3

Com a Mapper interna instalada, as linhas A15 e A14 passam a selecionar os registros da Mapper (R0 a R3) que por sua vez selecionam os blocos do banco de 256 Kb.

A15	A14	Registro
0	0	R0
0	1	R1
1	0	R2
1	1	R3

FUNCIONAMENTO

Na figura 1 temos o diagrama em blocos da Memory Mapper interna de 256 Kbytes. Nesta figura podemos notar que o circuito básico da mapper possui:

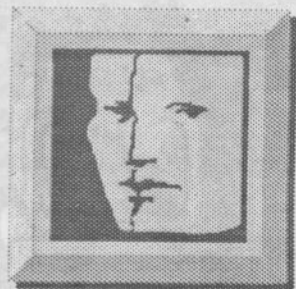
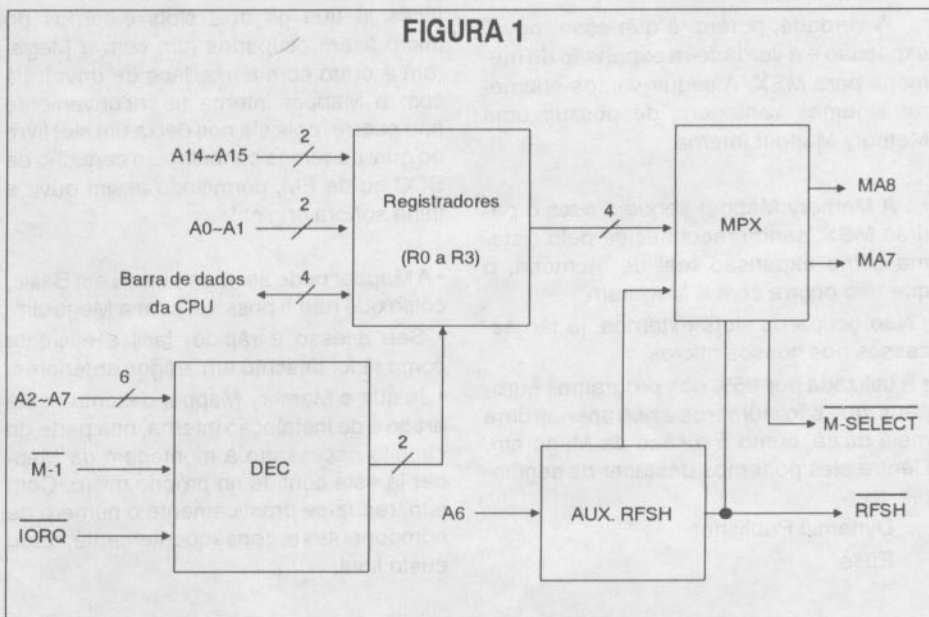
- um decodificador de I/O (Dec)
- quatro registradores (Regs)
- um multiplexador (MPX)
- um circuito auxiliar de Refresh (AuxRfsh)

O Z80 ao executar uma operação de leitura ou escrita (IN/OUT) nas portas de números &HFC a &HFF, faz com que o decodificador (Dec) envie um sinal de nível lógico "0" para os registros da Mapper, permitindo que estes sejam acessados pelo Z80.

Quando estes registros não estão sendo acessados pelo Z80 passam a ser selecionados pela linhas de endereços A15 a A14 e os dados contidos nestes são enviados ao multiplexador (MPX) que os transformam em linhas de endereçamento multiplexadas MA8 e MA7. Que também são parte integrante do circuito auxiliar de refresh.

O banco de 256 Kbytes é constituído por oito memórias dinâmicas (41256 ou equivalente). Estas memórias caracterizam-se pelo armazenamento temporário dos dados (por frações de segundos), sendo necessário que se faça periodicamente um "reavivamento" antes que os dados se percam. Para isto existe o Refresh.

O Z80 possui internamente um circuito de Refresh que provê um "reavivamento" de até 64Kbytes. O circuito auxiliar de refresh (AuxRfsh) faz com que este se estenda até os 256Kbytes.

Na próxima edição publicaremos a conclusão do projeto, com o esquema completo para você montar sua Memory Mapper. ❑

Batalha 1917

Um jogo completo para você digitar

Sérgio Cardoso Santos
Renato Ferreira Borges

O jogo Batalha 1917 foi elaborado pelos autores, em 1986, visando acrescentar mais uma opção aos apreciadores de jogos de estratégia. Na época a Gradiente chegou a analisar o programa, mas após a sua aprovação, surgiu a necessidade de ao menos um dos autores viajar a São Paulo a fim de demonstrar o programa, além de corrigir pequenos defeitos que ainda existiam. Infelizmente os respectivos cursos de engenharia freqüentados pelos autores os impediram de se ausentarem por um período maior; deste modo a idéia de comercialização foi relegada a um segundo plano.O programa é razoavelmente extenso, sendo constituído pela parte do programa em BASIC e pelos blocos em Linguagem de Máquina. O programa na sua forma integral continha tela de apresentação e voz. Entretanto, isto significaria mais 20K de linguagem de máquina para a digitação. Por este motivo, não serão apresentadas suas listagens.

A MONTAGEM DO JOGO

O programa BASIC da listagem 1 deve ser digitado e salvo em disquete com o nome de BAT.BAS, através do comando SAVE"BAT.BAS".

O programa da listagem 2, serve para auxiliar a digitação das rotinas em Linguagem de Máquina da listagem 3. Após digitar, salve este montador e execute-o.

Primeiro o programa pergunta o endereço inicial do bloco. Deve-se entrar o endereço inicial do bloco que está sendo digitado. Como exemplo, no caso do bloco BAT1.ASM, deve-se entrar E000 como endereço inicial. A partir daí deve-se digitar os códigos hexadecimais, conforme a listagem.

A cada 128 bytes digitados, o computador exibe na tela a soma parcial destes dados. Com isto é possível verificar se houve erro na digitação. Caso a soma não coincida com a apresentada na revista, tecle duas vezes ENTER e comece de novo (execute o programa novamente) a partir do endereço logo a seguir do último somatório. No fim do bloco tecle duas vezes ENTER e confira o somatório.

Quando todos os blocos já estiverem gravados com seus respectivos nomes, só resta entrar o programa BASIC abaixo. Digite-o, grave-o como "BAT.RUN" e execute-o. Bom divertimento.

```
10 KEYOFF : CLS
20 BLOAD"BAT1.ASM"
30 BLOAD"BAT2.ASM",R
40 BLOAD"BAT3.ASM",R
50 BLOAD"BAT4.ASM",R
60 RUN "BAT.BAS"
```

INSTRUÇÕES DO JOGO

OBJETIVO

O objetivo do jogo é simples e similar ao xadrez. Cada jogador tem como objetivo avançar seu exército em direção ao exército inimigo visando destruir o rei adversário.

EXÉRCITOS

Na parte superior da tela fica disposto o exército vermelho, enquanto na inferior o preto.

Cada exército é composto de 30 peças:

1 REI
5 CANHÕES
8 INFANTES (divididos em dois grupos de 4)
8 CAVALEIROS (divididos em dois grupos de 4)
8 TANQUES (divididos em dois grupos de 4)

As peças possuem uma hierarquia no que diz respeito a força em combates. Esta hierarquia é mostrada de forma crescente na lista acima, isto é, os tanques são as peças mais resistentes, enquanto o rei é a mais fraca.

CENÁRIO

O cenário é sorteado no início de cada jogo, o que impede que o jogo torne-se monótono, pois a cada nova partida diferentes estratégias deverão ser usadas. O cenário é constituído por lagos, florestas, minas e arames farpados.

LAGOS

Os lagos podem ser facilmente reconhecidos pela sua cor azul e o movimento de suas águas. Nenhuma peça consegue atravessá-los, devido à profundidade.

FLORESTAS

As florestas são constituídas por pinheiros verdes. Por serem florestas muito densas, só os tanques conseguem desbravá-las, abrindo caminho para que o resto do exército atravesse.

ARAMES FARPADOS

Os arames farpados são pretos e só podem ser atravessados por tanques.

MINAS

As minas (vestígios de guerras passadas) são losangos pretos espalhados pela tela. Evidentemente, todas as peças podem tentar atravessá-las, entretanto estarão correndo o risco que explodam. Devido ao tempo, muitas destas encontram-se desativadas, entretanto é impossível reconhecê-las sem experimentar.

BÚSSOLA

No centro da tela há um retângulo preto que deve ser usado como bússola durante o jogo. Nele encontram-se números que representam os oito pontos cardeais. Caso o jogador queira deslocar sua peça diagonalmente para cima e para direita (nordeste) deverá teclar o número 9 quando for solicitada a direção no rodapé da tela. Para facilitar as jogadas, a bússola tem o mesmo formato do teclado numérico de alguns MSX. Para não mover uma determinada peça tecla-se 0 na sua vez.

PEÇAS

Depois de responder a direção do movimento, o jogador deverá determinar a distância que a peça deve se mover. Evidentemente um cavaleiro tem uma facilidade de deslocar-se muito maior que a de um rei, portanto cada peça tem seu limite má-

ximo de deslocamento que é mostrado no rodapé da tela. Os limites são: Cavaleiro - 5, Infantes - 4, Tanques e Canhões - 3, Rei - 2. Os canhões são de grande valor para o jogador, pois além de movimentarem-se como as outras peças, possuem a capacidade de disparar sobre o exército inimigo, evidentemente com um certo limite de alcance (8). Na realidade, o disparos dos canhões nem sempre são precisos, portanto nunca confie exageradamente neles.

DUELOS

O exército preto subirá em direção ao vermelho e este descerá, de modo que em algum momento haverá o choque de peças. Neste choque, o computador emite sons que indicam o sucesso ou fracasso do ataque. Quando o som for agudo significará que a peça que atacou foi derrotada. Caso o sinal seja grave, esta peça foi a vencedora do duelo. A fim de dar um maior toque de realismo ao jogo, existe um desgaste das peças, isto é, num primeiro duelo é muito provável que um cavaleiro ganhe de um infante, entretanto neste duelo ele perde resistência, de modo que num outro, ele pode perder até de um rei. Existem ainda os soldados que se destacam na guerra, assim sendo é muito difícil (mas não impossível) que um infante ganhe de um cavaleiro de primeira.

SELECT

A tecla SELECT possibilita que o jogo seja salvo ou carregado (em disquete), que a jogada de uma determinada peça seja anulada ou ainda que a tela seja impressa. Evidentemente a impressora deve ser padrão. ❑

Sérgio Cardoso Santos. Cursa o 8º período de Engenharia de Sistemas e Computação na UERJ. Trabalha basicamente com linguagem C e Assembly. Tem conhecimento em Basic, Pascal, C, Assembly, Dbase, Lotus, CAD, DOS e Windows. Possui um MSX 2 e um AT386 SX25, com SVGA, impressora e modem.

Renato Ferreira Borges. Cursa o 9º período de Engenharia de Computação na PUC. Tem conhecimento em Basic, Pascal, C, C++, Assembly, Prolog, Lisp, Windows, DOS, Unix e Motif. Possui um MSX 1.1 e um AT386 DX33 com SVGA, impressora e modem.

LISTAGEM DO JOGO

```
10 ' ------------------------------------------------
11 ' ---------------- BATALHA 1917 ------------------
12 ' -------- PROGRAMA DESENVOLVIDO POR : -----------
13 ' ---------- SERGIO CARDOSO SANTOS ---------------
14 ' --------- RENATO FERREIRA  BORGES --------------
15 ' ------------------------------------------------
20 SCREEN,,0:ONSTOPGOSUB65528:STOPON:ONERRORGOTO6
5527:POKE&HFBB1,0:POKE&HFCAB,1:DEFUSR1=&HE084:VPO
KEBASE(5)+705,189
30 ONINTERVAL=10GOSUB10000:DEFINTA-Z:DIMZ(78),X(7
8),G(78):A$=CHR$(&HD8)
60 A=INT(RND(1)*26)+1:Z=A
70 B=INT(RND(-TIME)*12)+5:X=B
80 C=INT(RND(1)*5)+3
90 D=INT(RND(-TIME)*7)+3
100 E=INT(RND(-TIME)*2)+1
110 F=INT(RND(-TIME)*2)+1
120 VPOKE339,248
130 FORXX=1TOD
140 FORZZ=1TOC
150 IFB>16ORB<6 OR Z<1OR Z>29THEN 190
160 LOCATEZ,X:PRINTA$
170 IFE=1THENZ=Z+1ELSEZ=Z-1
180 NEXTZZ
190 IFF=1THENB=B+1ELSEB=B-1
200 Z=A:X=B:NEXTXX
210 VV=VV+1:IFVV<10ANDA$=CHR$(&HD8) THEN60
220 A$=CHR$(&HE0):IFVV<15 THEN60
230 A$=CHR$(&HE8)
240 FORC=1TO15
250 A=INT(RND(1)*28)+1
260 B=INT(RND(1)*21)+1
270 LOCATEA,B:PRINTA$
280 NEXTC
290 IFA$=CHR$(&HE8)THENA$=CHR$(&HE9):GOTO240
300 IFA$=CHR$(&HE9)THENA$=" ":GOTO240
310 IFA$=" "THENA$=" ":GOTO240
320 LOCATE12,9:PRINTCHR$(&HE0);CHR$(&HE0);CHR$(&H
E0);CHR$(&HE0);CHR$(&HE0):LOCATE12,13:PRINTCHR$(&
HE0);CHR$(&HE0);CHR$(&HE0);CHR$(&HE0);CHR$(&HE0):
LOCATE12,10:PRINTCHR$(&HE0);CHR$(&H96);CHR$(&H97)
;CHR$(&H98);CHR$(&HE0)
321 LOCATE12,11:PRINTCHR$(&HE0);CHR$(&H93);CHR$(&
H94);CHR$(&H95);CHR$(&HE0):LOCATE12,12:PRINTCHR$(&
HE0);CHR$(&H90);CHR$(&H91);CHR$(&H92);CHR$(&HE0)
330 '
340 Z(0)=16+32:Z(2)=45+32:Z(3)=44+32:Z(4)=13+32:Z
(5)=12+32:Z(7)=52+32:Z(8)=51+32:Z(9)=20+32:Z(10)=
19+32:Z(12)=41+32:Z(13)=40+32:Z(14)=9+32:Z(15)=8+
32:Z(17)=56+32:Z(18)=55+32:Z(19)=24+32:Z(20)=23+3
2:Z(22)=37+32:Z(23)=36+32:Z(24)=5+32:Z(25)=4+32
350 Z(27)=60+32:Z(28)=59+32:Z(29)=28+32:Z(30)=27+
32:Z(32)=110+32:Z(33)=111+32:Z(34)=112+32:Z(35)=1
13+32:Z(36)=114+32:Z(38)=720-32:Z(40)=684-32:Z(41)
=685-32:Z(42)=716-32:Z(43)=717-32:Z(45)=691-32:Z(
46)=692-32:Z(47)=723-32:Z(48)=724-32:Z(50)=680-32
360 Z(51)=681-32:Z(52)=712-32:Z(53)=713-32:Z(60)=
676-32:Z(61)=677-32:Z(62)=708-32:Z(63)=709-32:Z(6
5)=699-32:Z(66)=700-32:Z(67)=731-32:Z(68)=732-32:
Z(55)=695-32:Z(56)=696-32:Z(57)=727-32:Z(58)=728-
32:Z(70)=622-32:Z(71)=623-32:Z(72)=624-32:Z(73)=6
25-32
361 A=RND(1)*10-5:X(0)=7+A:X(38)=7+A:FORA=2TO5:B=
RND(1)*10-5:X(A+43)=25+B:X(A)=25+B:NEXTA:FORA=40T
O43:B=RND(1)*10-5:X(A)=25+B:X(A-33)=25+B:NEXTA:FO
RA=12TO15:B=RND(1)*10-5:X(A)=20+B:X(A+43)=20+B:NE
XTA:FORA=50TO53:B=RND(1)*10-5:X(A)=20+B:X(A-33)=2
0+B:NEXTA
362 FORA=60TO63:B=RND(1)*10-5:X(A)=15+B:X(A-33)=1
5+B:NEXTA:FORA=32TO36:B=RND(1)*10-5:X(A)=10+B:X(A
+38)=10+B:NEXTA:Z(74)=626-32:FORA=22TO25:B=RND(1)*
10-5:X(A)=15+B:X(A+43)=15+B:NEXT
363
FORA=0TO78:G(A)=6:NEXTA:RESTORE363:DATA1,6,11,16,2
1,26,31,37,39,44,49,54,59,64,69,75
364 FORA=1TO16:READB:Z(B)=-1:X(B)=0:NEXTA
370 GOSUB380:GOSUB390:GOSUB400:GOSUB410:GOSUB420:
GOSUB430:GOSUB435:GOSUB440:GOSUB450:GOSUB520:GOSU
B460:GOSUB470:GOSUB480:GOSUB490:GOSUB500:GOSUB510
:INTERVALON:GOTO1000
380 IFX(0)>0ANDX(0)<31THENVPOKEBASE(5)+Z(0),&HF0:
RETURN390 FORA=2TO5:IFX(A)>0ANDX(A)<31THENVPOKEBA
SE(5)+Z(A),&HF1:NEXTA:RETURNELSENEXTA:RETURN
400 FORA=7TO10:IFX(A)>0ANDX(A)<31THENVPOKEBASE(5)
+Z(A),&HF5:NEXTA:RETURNELSENEXTA:RETURN
410 FORA=12TO15:IFX(A)>0ANDX(A)<31THENVPOKEBASE(5
)+Z(A),&HF2:NEXTA:RETURNELSENEXTA:RETURN
420 FORA=17TO20:IFX(A)>0ANDX(A)<31THENVPOKEBASE(5
)+Z(A),&HF6:NEXTA:RETURNELSENEXTA:RETURN
430 FORA=22TO25:IFX(A)>0ANDX(A)<31THENVPOKEBASE(5
)+Z(A),&HF3:NEXTA:RETURNELSENEXTA:RETURN
435 FORA=27TO30:IFX(A)>0ANDX(A)<31THENVPOKEBASE(5
)+Z(A),&HF7:NEXTA:RETURNELSENEXTA:RETURN
440 FORA=32TO36:IFX(A)>0ANDX(A)<31THENVPOKEBASE(5
)+Z(A),&HF4:NEXTA:RETURNELSENEXTA:RETURN
450 IFX(38)>0ANDX(38)<31THENVPOKEBASE(5)+Z(38),&H
F8
460 FORA=40TO43:IFX(A)>0ANDX(A)<31THENVPOKEBASE(5
)+Z(A),&HF9:NEXTA:RETURNELSENEXTA:RETURN
470 FORA=45TO48:IFX(A)>0ANDX(A)<31THENVPOKEBASE(5
)+Z(A),&HFD:NEXTA:RETURNELSENEXTA:RETURN
```

```
480 FORA=50TO53:IFX(A)>0ANDX(A)<31THENVPOKEBASE(5
)+Z(A),&HFA:NEXTA:RETURNELSENEXTA:RETURN
490 FORA=55TO58:IFX(A)>0ANDX(A)<31THENVPOKEBASE(5
)+Z(A),&HFE:NEXTA:RETURNELSENEXTA:RETURN
500 FORA=60TO63:IFX(A)>0ANDX(A)<31THENVPOKEBASE(5
)+Z(A),&HFB:NEXTA:RETURNELSENEXTA:RETURN
510 FORA=65TO68:IFX(A)>0ANDX(A)<31THENVPOKEBASE(5
)+Z(A),&HD0:NEXTA:RETURNELSENEXTA:RETURN
520 FORA=70TO74:IFX(A)>0ANDX(A)<31THENVPOKEBASE(5
)+Z(A),&HFC:NEXTA:RETURNELSENEXTA:RETURN
1000 '
1010 IF YY=0THENXX$=CHR$(&HF0)ELSEXX$=CHR$(&HF8)
1020 GOSUB2890:I=2:GOSUB 2600
1030 IF G=0THEN1090
1040 RR=Z(YY):FORY=1TOU
1050 GOSUB2660:IFRR<0ORRR>735THEN1090
1060 IFCH<>32ANDCH<>&HE9THENWW=0:GOSUB60000:IFWW=
1THEN1090
1061 IFRR>704THEN1090
1070 POKEBASE(5)+Z(YY),32:Z(YY)=RR
1071 A=USR1(2):GOSUB25030:IFCH=&HE9THENGOSUB65500
1075 IFYY=0THENGOSUB380ELSEGOSUB450
1080 NEXTY
1090 GOSUB2890
1100 '
1109 YY=YY+2
1110 FORA=YYTOYY+3:IF X(A)=31THENNEXTA:YY=YY+3:GO
TO1280
1120 IFYY<6THENXX$=CHR$(&HF1)ELSEIFYY<11THENXX$=C
HR$(&HF5)ELSEIFYY<44THENXX$=CHR$(&HF9)ELSEXX$=CHR$
(&HFD)
1130 GOSUB2890
1131 I=3:GOSUB 2600
1140 IF G=0THENYY=YY+3:GOTO1270
1150 TT=2:GOSUB50000
1220 RR=Z(YY):FORY=1TOU:GOSUB2660:IFRR<0ORRR>735T
HEN1260
1230 IFCH<>32 ANDCH<>&HD8 ANDCH<>&HE8 ANDCH<>&HE9
THENWW=0:GOSUB60000:IFWW=1THEN1260
1231 IFRR>704THEN1260
1240 IFX(YY)>0ANDX(YY)<31THENA=USR1(2):GOSUB25030
:VPOKEBASE(5)+Z(YY),32:Z(YY)=RR:IFYY<6THENGOSUB39
0ELSEIFYY<11THENGOSUB400ELSEIFYY<44THENGOSUB460EL
SEGOSUB470
1241 IF CH=&HE9THENGOSUB65500
1250 NEXTY
1260 IF X(YY+1)<>0THENYY=YY+1:GOTO1220
1270 GOSUB2890
1280 IFYY<=5THENYY=7:GOTO1110
1290 IFYY<=43ANDYY>39THENYY=45:GOTO1110
1300 '
1309 YY=YY+2
1310 FORA=YYTOYY+3:IFX(A)=31THENNEXTA:YY=YY+3:GOT
O1490
1320 IFYY<16THENXX$=CHR$(&HF2)ELSEIFYY<21THENXX$=
CHR$(&HF6)ELSEIFYY<54THENXX$=CHR$(&HFA)ELSEXX$=CH
R$(&HFE)
1330 GOSUB2890
1331 I=5:GOSUB 2600
1340 IF G=0THENYY=YY+3:GOTO1480
1350 TT=2:GOSUB50000
1420 RR=Z(YY):FORY=1TOU
1430 GOSUB2660:IFRR<0ORRR>735THEN1470
1440 IFCH<>32ANDCH<>&HE9THENWW=0:GOSUB60000:IFWW=
1THEN1470
1441 IFRR>704THEN1470
1450 IFX(YY)>0ANDX(YY)<31THENA=USR1(2):GOSUB25030
:VPOKEBASE(5)+Z(YY),32:Z(YY)=RR:IFYY<16THENGOSUB4
10ELSEIFYY<21THENGOSUB420ELSEIFYY<54THENGOSUB480E
LSEGOSUB490
1451 IF CH=&HE9THENGOSUB65500
1460 NEXTY
1470 IF X(YY+1)<>0THENYY=YY+1:GOTO1420
1480 GOSUB2890
1490 IFYY<=15THENYY=17:GOTO1310
1500 IFYY<=53ANDYY>49THENYY=55:GOTO1310
1510 '
1519 YY=YY+2
1520 FORA=YYTOYY+3:IF X(A)=31THENNEXTA:YY=YY+3:GO
TO1700
1530 IF YY<26THENXX$=CHR$(&HF3)ELSEIFYY<31THENXX$
=CHR$(&HF7)ELSEIFYY<64THENXX$=CHR$(&HFB)ELSEXX$=C
HR$(&HD0)
1540 GOSUB2890
1541 I=4:GOSUB 2600
1550 IF G=0THENYY=YY+3:GOTO1690
1560 TT=2:GOSUB50000
1630 RR=Z(YY):FORY=1TOU
1640 GOSUB2660:IFRR<0ORRR>735THEN1680
1650 IFCH<>32ANDCH<>&HE9 THENWW=0:GOSUB60000:IFWW
=1THEN1680
1651 IFRR>704THEN1680
1660 IFX(YY)>0ANDX(YY)<31THENA=USR1(2):GOSUB25030
:VPOKEBASE(5)+Z(YY),32:Z(YY)=RR:IFYY<26THENGOSUB4
30ELSEIFYY<31THENGOSUB435ELSEIFYY<65THENGOSUB500E
LSEGOSUB510
1661 IF CH=&HE9THENGOSUB65500
1670 NEXTY
1680 IF X(YY+1)<>0THENYY=YY+1:GOTO1630
1690 GOSUB2890
1700 IFYY<=25THENYY=27:GOTO1520
1710 IFYY<=63ANDYY>59THENYY=65:GOTO1520
1720 '
1730 YY=YY+2:FORA=YYTOYY+4:IF X(A)=31THENNEXTA:YY
=YY+4:SS=SS+4:GOTO 2060
1740 IF YY<37THENXX$=CHR$(&HF4)ELSEXX$=CHR$(&HFC)
1750 GOSUB2890
1751 I=3:GOSUB 2600
1760 IF G=0THENYY=YY+4:GOTO1870
1770 TT=3:GOSUB50000
1810 RR=Z(YY):FORY=1TOU
1820 GOSUB2660:IFRR<0ORRR>735THEN1860
1830 IFCH<>32ANDCH<>&HE9 THENWW=0:GOSUB60000:IFWW
=1THEN1860
1831 IFRR>704THEN1860
1840 IFX(YY)>0ANDX(YY)<31THENA=USR1(2):GOSUB25030
:VPOKEBASE(5)+Z(YY),32:Z(YY)=RR:IFYY<37THENGOSUB4
40ELSEGOSUB520
1841 IF CH=&HE9THENGOSUB65500
1850 NEXTY
1860 IF X(YY+1)<>0THENYY=YY+1:GOTO1810
1870 GOSUB2890:GOSUB2890
1880 FORA=YY-4TOYY:IFX(A)=31THENNEXTA:GOTO2060ELS
ELOCATE0,22:PRINT"Direção";CHR$(&HBD);"do";CHR$(&
HBD);"tiro";CHR$(&HBD);"?";CHR$(&HBD);CHR$(&HBD);
CHR$(&HBD);CHR$(&HBD);CHR$(&HBD);CHR$(&HBD);CHR$(&
HBD);CHR$(&HBD);CHR$(&HBD);CHR$(&HBD);CHR$(&HBD);
1890 SS=YY-4:PL=0:LP=1:GOSUB30000:IF(ASC(I$)>&H2F
ANDASC(I$)<&H3A ANDASC(I$)<>&H35)THENG=VAL(I$)ELS
E1880
1900 IF G=0THENSS=SS+4:GOTO2050
1901 IF (G=7ORG=4ORG=1ORG=8)AND(G(SS)=9ORG(SS)=6O
RG(SS)=3ORG(SS)=2)THENGOSUB65100:GOTO 1909
1902 IF (G=9ORG=6ORG=3ORG=2)AND(G(SS)=1ORG(SS)=4O
RG(SS)=7ORG(SS)=8)THENGOSUB65100
1909 FORA=SSTOSS+4:G(A)=G:NEXTA
1910 LOCATE0,22:PRINT"Alcance";CHR$(&HBD);"?";CHR
$(&HBD);CHR$(&HBD);"1-8";CHR$(&HBD);CHR$(&HBD);CH
R$(&HBD);CHR$(&HBD);CHR$(&HBD);CHR$(&HBD);CHR$(&H
BD);CHR$(&HBD);CHR$(&HBD);CHR$(&HBD);CHR$(&HBD);C
HR$(&HBD);
1915 PL$=CHR$(ASC(MID$(STR$(G),2,2))+&H5F)
1920 LP=1:PL=1:GOSUB30000:U=VAL(I$):IFU<1ORU>8THE
N1910
1930 RR=Z(SS):FORA=1TOU:IFX(SS)=31THEN2040
1940 GOSUB2660
1943 IFRR/32=(RR\32)THENGOSUB3920:GOTO 2040
1944  IFRR+1/32=(RR+1\32)THENGOSUB3920:GOTO 2040
1950 NEXTA
1960 L=INT(RND(-TIME)*8):IFG<>4ANDG<>6THENIFL=2TH
ENL=32ELSEIFL=3THENL=-32ELSEL=0
1965 IFG=4ORG=6THENIFL=2THENL=-1ELSEIFL=3THENL=1E
LSEL=0
1970 RR=RR+L:CH=VPEEK(BASE(5)+RR)
1971 IF (CH>&H8F ANDCH<&H9A)ORCH=&HDB OR (CH>&H2C
ANDCH<122)ORCH=189THENGOSUB3920:GOTO 2040
1980 VPOKEBASE(5)+RR,&HF:IFCH=&HE0THENA=USR1(2):G
OSUB25020ELSEA=USR6(0):GOSUB3920
1990 IFRR<0ORRR>704THEN2040
1999 IF CH<&HF0 ANDCH<>&HD0THEN2001
```

MSX PageMaker DeLuxe

A Nemesis acaba de provar mais uma vez o seu amor pelo MSX...

Ela está lançando com exclusividade, a grande novidade do ano em termos de software para esta linha. Trata-se da nova versão do maior "best-seller" nacional, o "MSX PAGE MAKER".

É o "MSX PAGE MAKER DeLuxe" que incorpora diversas implementações inéditas no mercado, tornando-o comparável aos melhores produtos internacionais criados para as linhas Amiga e PC! Veja alguns de seus recursos:

Maior velocidade de operação; instalação automática com ou sem "MEGARAM DISK"; novos módulos para confecção de faixas gigantes e posters de criação imediata; novas e mais completas funções de desenho; novas opções de bordas de página com possibilidade de redefinição e carregamento de bordas personalizadas; novo sistema de manipulação de "shapes" com criação, rotação, reversão, inédito movimento ponto-a-ponto.

O novo "MSX PAGE MAKER DeLuxe" importa textos criados em outros editores como o "MSX WRITE" ou o "WORDSTAR", com filtro de acentuação e caracteres de contrôle. Importa também toda a variedade de fontes de alfabetos, figuras e "Clip-Arts" existentes para a linha MSX e PC.

Sim, o "MSX PAGE MAKER DeLuxe" lê diretamente discos de letras e figuras do PC no padrão "PRINT MASTER", "PRINT SHOP" e "NEWS MASTER". Acredite se quiser!

As páginas criadas no "MSX PAGE MAKER DeLuxe" podem ser gravadas em formato compactado, o que economiza espaço nos disquetes. Podem ser impressas em 3 tipos de resolução, inclusive em resolução de "plotter". As páginas podem ser impressas também através micros da linha PC.

Se você já possui uma cópia original do "MSX PAGE MAKER", você não pode deixar de atualizar a sua versão por uma pequena parcela do valor de tabela. Mas se você ainda não possui um "MSX PAGE MAKER", aproveite para entrar no fascinante mundo do "desk-top publishing" com seu MSX. Você verá que nunca foi tão fácil criar cartazes, trabalhos escolares, cartões pessoais e festivos, propagandas em geral, estórias em quadrinhos, "fanzines", etc.

O "MSX PAGE MAKER DeLuxe" funciona em disk-drives de 5 1/4 ou 3 1/2 conectados a qualquer MSX nacional ou importado, e vem com um completo manual de instruções.

Preço de lançamento: Cr$ 250.000,00 em 5 1/4 e Cr$ 280.000,00 em 3 1/2. Se você já possui uma versão anterior, envie o disco com a mesma, e inclua um cheque de Cr$ 150.000,00 para a NEMESIS INFORMÁTICA LTDA. no endereço: Caixa postal 4.583 Cep 20.001-970 - Rio de Janeiro - RJ. Ou venha diretamente ao nosso "Show-Room" na Rua Sete de Setembro, 92 sala 1.203 - Centro - Rio de Janeiro - RJ.

```
2000 FORA=0TO75:IFRR=Z(A)THENZ(A)=-1:X(A)=31:PL$=
CHR$(&HF):SWAPXX$,PL$:FORR=1TO24:GOSUB2890:NEXTR:
SWAPPL$,XX$:GOTO 2001ELSE NEXTA
2001 IF X(0)=31ORX(38)=31THEN60100
2020 VPOKEBASE(5)+RR,32
2030 IF CH=&HE0 THENVPOKEBASE(5)+RR,CH
2040 IFX(SS+1)<>0THENSS=SS+1:GOTO1930
2050 GOSUB2890
2060 SS=SS+38:YY=YY+2:IFYY>73THENYY=0
2061 GOTO 1000
2600 LOCATE0,22:PRINT"Direcao";CHR$(&HBD);"?";CHR
$(&HBD);CHR$(&HBD);CHR$(&HBD);CHR$(&HBD);CHR$(&HB
D);CHR$(&HBD);CHR$(&HBD);CHR$(&HBD);CHR$(&HBD);CH
R$(&HBD);CHR$(&HBD);CHR$(&HBD);CHR$(&HBD);CHR$(&H
BD);CHR$(&HBD);CHR$(&HBD);CHR$(&HBD);CHR$(&HBD);
2610 LP=0:PL=0:GOSUB30000:IF(ASC(I$)>&H2FANDASC(I
$)<&H3A ANDASC(I$)<>&H35)THENG=VAL(I$)ELSE2600
2620 IF G=0THENRETURN
2625 IF (G=7ORG=4ORG=1ORG=8)AND(G(YY)=9ORG(YY)=6O
RG(YY)=3ORG(YY)=2)THENGOSUB65100:GOTO 2629
2626 IF (G=9ORG=6ORG=3ORG=2)AND(G(YY)=1ORG(YY)=4O
RG(YY)=7ORG(YY)=8)THENGOSUB65100
2629 IF YY=0ORYY=38THENG(YY)=GELSEFORA=YYTOYY+4:G
(A)=G:NEXTA
2630 LOCATE0,22:PRINT"Distância";CHR$(&HBD);"?";C
HR$(&HBD);CHR$(&HBD);"MAX>";MID$(STR$(I),2,1);CHR$
(&HBD);CHR$(&HBD);CHR$(&HBD);CHR$(&HBD);CHR$(&HBD)
;CHR$(&HBD);CHR$(&HBD);CHR$(&HBD)
2639 PL$=CHR$(ASC(MID$(STR$(G),2,2))+&H5F):PL=1
2640 LP=0:GOSUB30000:U=VAL(I$):IFU<1ORU>ITHEN2630
2650 RETURN
2660 IFG=8THENRR=RR-32:RT=-32
2670 IFG=9THENRR=RR-31:RT=-31
2680 IFG=6THENRR=RR+1:RT=1
2690 IFG=3THENRR=RR+33:RT=33
2700 IFG=2THENRR=RR+32:RT=32
2710 IFG=1THENRR=RR+31:RT=31
2720 IFG=4THENRR=RR-1:RT=-1
2730 IFG=7THENRR=RR-33:RT=-33
2740 CH=VPEEK(BASE(5)+RR)
2750 RETURN
2890 C$="000"+HEX$(8*ASC(XX$)):B$=RIGHT$(C$,4):PO
KE&HE0CB,VAL("&H"+RIGHT$(B$,2)):POKE&HE0CC,VAL("&
H"+LEFT$(B$,2)):U1=USR1(1):RETURN
2900 KS=31:GOSUB2890:A=USR1(2)
2901 ZA=1:A=0:B=13:D=0:E=13:CLS:IFX(0)=31ANDX(38)
=31THEN65000
2905 IF X(0)=31THENV=2:GOTO2930
2910 IF X(38)=31THENV=1:GOTO2930
2920 IF YY=0 THEN V=1ELSE V=2
2930 CLS:LOCATE3,3:PRINT"O EXERCITO VENCEDOR E O"
2940 LOCATE10,7:IFV=1THENPRINT"VERMELHO"ELSEPRINT
" PRETO":GOTO3010
2950 G$="":FORC=2TO10:IFX(C)>0ANDX(C)<31THENG$=G$
+CHR$(&HF1):NEXTCELSENEXTC
2960 FORC=12TO20:IFX(C)>0ANDX(C)<31THENG$=G$+CHR$
(&HF2):NEXTCELSENEXTC
2970 FORC=22TO30:IFX(C)>0ANDX(C)<31THENG$=G$+CHR$
(&HF3):NEXTCELSENEXTC
2980 FORC=32TO36:IFX(C)>0ANDX(C)<31THENG$=G$+CHR$
(&HF4):NEXTCELSENEXTC
2990 A=(29-LEN(G$))/2:LOCATE13,B-2:PRINTCHR$(&HF0
):LOCATEA,B:PRINTG$:IFZA=1THENE=E+5:ZA=ZA+2
3000 IFZA=2THEN 3060
3010 G$="":FORC=40TO48:IFX(C)>0ANDX(C)<31THENG$=G
$+CHR$(&HF9):NEXTCELSENEXT
3020 FORC=50TO58:IFX(C)>0ANDX(C)<31THENG$=G$+CHR$
(&HFA):NEXTCELSENEXTC
3030 FORC=60TO68:IFX(C)>0ANDX(C)<31THENG$=G$+CHR$
(&HFB):NEXTCELSENEXTC
3040 FORC=70TO74:IFX(C)>0ANDX(C)<31THENG$=G$+CHR$
(&HFC):NEXTCELSENEXTC
3050 D=(29-LEN(G$))/2:LOCATE13,E-2:PRINTCHR$(&HF8
):LOCATED,E:PRINTG$:IFZA=1THENZA=ZA+1:B=B+5:GOTO2
950
3051 IFX(0)=31ANDX(38)=31THEN65010
3060 FORA=1TO10000:NEXTA
3870 CLS:LOCATE4,9:PRINT"OUTRO JOGO ? (S/N)"
3880 A=USR1(4):SCREEN,,1
3890 A$=INKEY$:IF A$=""THEN3890
3900 SCREEN,,0:IF A$="S"THENCLEAR:CLS:FORS=1TO23:
VPOKEBASE(5)+G,&HDB:G=G+32:VPOKEBASE(5)+G-1,&HDB:
NEXTS:FORS=1TO32:VPOKEBASE(5)+S,&HDB:VPOKEBASE(5)+
S+735,&HDB:NEXTS:RUN
3910 END
3920 RESTORE3930:FOR TY=0TO13:READB:SOUNDTY,B:NEX
TTY:RETURN
3930 DATA 0,0,0,0,0,0,20,247,16,0,0,100,60,0
10000 DEFUSR=&HE041:Y9=USR(0):RETURN
25000 SOUND11,100:SOUND12,150:SOUND2,100:SOUND3,1
5:SOUND9,15:FORA=1TO100:NEXTA:SOUND9,0
25001 SOUND0,255:SOUND1,5:SOUND2,255:SOUND3,8:SOU
ND4,0:SOUND5,0:SOUND6,0:SOUND7,&B111000:SOUND8,16
:SOUND9,16:SOUND10,16:SOUND13,0:FORTY=1TO390:NEXT
TY:RETURN
25010 SOUND11,100:SOUND12,150:SOUND2,100:SOUND3,1
5:SOUND9,15:FORA=1TO100:NEXTA:SOUND9,0
25011 SOUND0,255:SOUND1,0:SOUND2,0:SOUND3,8:SOUND
4,0:SOUND5,0:SOUND6,0:SOUND7,&B111000:SOUND8,16:S
OUND9,16:SOUND10,16:SOUND13,0:FORTY=1TO400:NEXTTY
:RETURN
25020 SOUND11,20:SOUND12,35:SOUND7,0:SOUND6,10:SO
UND8,16:SOUND9,16:SOUND10,16:SOUND13,0:VPOKEBASE(
5)+RR,CH:RETURN
25030 SOUND2,25:SOUND3,10:SOUND9,15:FORA=1TO20:NE
XTA:SOUND9,0:RETURN
30000 A=USR1(4):B=1:C=1
30010 I$=INKEY$:SCREEN,,1
30015 GOSUB 2890:B=B+1
30020 IFI$=""THENFORA=1TO90:NEXTA:IFPL=0THENGOTO3
0010ELSESWAPXX$,PL$:GOSUB2890:C=C+1:SWAPXX$,PL$:G
OTO 30010
30025 SCREEN,,0:IF B/2=B\2THENGOSUB2890
30026 IF C/2=C\2THENSWAPXX$,PL$:GOSUB2890:SWAPXX$
,PL$
30030 IFI$=CHR$(&H18)THENRETURN40000
30040 RETURN
35000 LOCATE0,22:PRINT"(I)mprime";CHR$(&HBD);"(A)
borta";CHR$(&HBD);"(M)enu";CHR$(&HBD);"1";CHR$(&H
BD);:A=USR1(4):SCREEN,,1
35001 I$=INKEY$:IFI$=""THENFORA=1TO40:NEXTA:GOTO3
5001
35002 SCREEN,,0:IFI$="A"THENIFLP=0THEN2600ELSE188
0
35003 IFI$="I"THENPU=1:A=USR1(7):GOTO 60104
35004 IFI$<>"M"THEN35000
40000 LOCATE0,22:PRINT"(L)oad";CHR$(&HBD);"(S)ave
";CHR$(&HBD);"(M)enu";CHR$(&HBD);"2";CHR$(&HBD);C
HR$(&HBD);CHR$(&HBD);CHR$(&HBD);CHR$(&HBD);CHR$(&
HBD);:A=USR1(4):SCREEN,,1
40001 I$=INKEY$:IFI$=""THENFORA=1TO40:NEXTA:GOTO4
0001
40002 SCREEN,,0:IFI$="L"THEN 40060
40010 IFI$="S"THEN40030
40020 IFI$="M"THEN35000
40021 GOTO 40000
40030 BSAVE"B1.DAT",6144,6911,S
40040 OPEN"B2.DAT"FOR OUTPUTAS#1
40050 FORA=0TO75::PRINT#1,X(A):PRINT#1,Z(A):NEXTA
:PRINT#1,YY:PRINT#1,SS:CLOSE:IFLP=0THENGOTO2600EL
SE1880
40060 GOSUB2890:BLOAD"B1.DAT",S
40070 OPEN"B2.DAT"FOR INPUTAS#1
40080 FORA=0TO75:INPUT#1,X(A):INPUT#1,Z(A):NEXTA:
INPUT#1,YY:INPUT#1,SS:CLOSE
40090 IFYY>37THENKL=YY-38ELSEKL=YY
40100 ONKL+1GOTO1000,0,1120,1120,1120,1120,0,1120
,1120,1120,1120,0,1320,1320,1320,1320,0,1320,1320,
1320,1320,0,1530,1530,1530,1530,0,1530,1530,1530,1
530,0,1740,1740,1740,1740,1740
50000 IF G=7ORG=8ORG=9ORG=4THENFORJ=1TO4:FORK=YYT
OYY+TT:IFZ(K)>Z(K+1)THENSWAPZ(K),Z(K+1):SWAPX(K),
X(K+1):NEXTK,JELSENEXTK,J
50010 IF G=6ORG=3ORG=2ORG=1THENFORJ=1TO4:FORK=YYT
OYY+TT:IFZ(K)<Z(K+1)THENSWAPZ(K),Z(K+1):SWAPX(K),
X(K+1):NEXTK,JELSENEXTK,J
50020 RETURN
60000 '
60001 IF CH=&HDBOR CH=&HE0 OR CH=&HE8 OR CH=&HD8T
HEN WW=1:RETURN
60010 IF YY<37 ANDCH<&HF8 AND CH>&HEFTHENWW=1:RET
URN
60020 IF YY>37 AND(CH<&HFFANDCH>&HF7ORCH=&HD0)THE
NWW=1:RETURN
```

```
60025 IFYY<37THENFF=38ELSEFF=0
60030 FORI=FFTOFF+37:IFRR=Z(I)THENE8=X(YY):X(YY)=
X(YY)-X(I):X(I)=X(I)-E8ELSENEXTI
60040 IFX(YY)<1THENX(YY)=31:A=USR1(2):GOSUB25010:
VPOKEBASE(5)+RR-RT,32:Z(YY)=-1:WW=1
60050IFX(I)<1THENA=USR1(2):GOSUB25000:X(I)=31:Z(I)
=-1:VPOKEBASE(5)+RR,32
60051 IF X(0)=31ORX(38)=31THEN 60100
60060 RETURN
60100 A=USR1(4)
60101 LOCATE0,22:PRINT"Quer imprimir ? (S/N)";:SC
REEN,,1:I$=INKEY$:IFI$=""THEN60101
60102 SCREEN,,0:IF I$="N"THEN2900
60103 IF I$<>"S"THEN60101
60104 LOCATE0,22:PRINT"Batalha 1917  -  by S&RSOF
T"
60200 POKE&HE0DF,&H6E:POKE&HE0E0,&H79:A=USR1(3):I
FPU=0THEN2900ELSEA=USR1(8):GOSUB65529:PU=0:IFLP=0
THEN2600ELSE1880
65000 LOCATE11,3:PRINT"EMPATE":GOTO 2950
65010 YY=0:Z=15:LOCATE4,22:PRINT"DECISAO POR PONT
OS !":FORA=1TO1000:NEXT
65015 LOCATE1,Z:PRINT"PEÇA =":LOCATE15,Z:PRINT"TO
TAL="
65017 IF YY>1ANDYY<11THENXX$=CHR$(&HF1)ELSEIFYY>3
9ANDYY<49THENXX$=CHR$(&HFD)
65018 IF YY>11ANDYY<21THENXX$=CHR$(&HF2)ELSEIFYY>
49ANDYY<59THENXX$=CHR$(&HFE)
65019 IF YY>21ANDYY<37THENXX$=CHR$(&HF3)ELSEIFYY>
59ANDYY<69THENXX$=CHR$(&HD0)
65020
IFYY=0THENXX$=CHR$(&HF0)ELSEIFYY=38THENXX$=CHR$(&
HF8)
65021 IF YY>31ANDYY<37THENXX$=CHR$(&HF4)ELSEIFYY>
69ANDYY<75THENXX$=CHR$(&HFC)

65025 IFX(YY)<31ANDX(YY)>0THENLOCATE8,Z:PRINTXX$:
LOCATE22,Z:SO=SO+X(YY):PRINTSO

65027 YY=YY+1:IFYY=37THENZ=B+7:DO=SO:SO=0

65028 IFYY>=75THEN65030ELSEGOTO65015

65029 IF SO>DOTHENV=2ELSEV=1

65030 IF SO=DOTHENLOCATE4,22:PRINT"          EMPATE
":GOTO3060

65031 LOCATE3,3:PRINT"O EXERCITO VENCEDOR E O"

65040
LOCATE10,7:IFV=1THENPRINT"VERMELHO"ELSEPRINT" PRE
TO":GOTO3060

65100 JH$="000"+HEX$(8*ASC(XX$)):JH$=RIGHT$(JH$,4
):POKE&HE001,VAL("&H"+RIGHT$(JH$,2)):POKE&HE002,V
AL("&H"+LEFT$(JH$,2)):A8=USR1(5):RETURN

65500 A!=RND(-TIME):IFA!<.5THENRETURNELSEX(YY)=31
:Z(YY)=-1:VPOKEBASE(5)+RR,32:VPOKEBASE(5)+RR-RT,3
2:A=USR1(2):GOSUB3920:VPOKEBASE(5)+RR,&HD7:PL$=CH
R$(&HD7):SWAPXX$,PL$:FORB=1TO10:GOSUB2890:NEXTB:S
WAPXX$,PL$:VPOKEBASE(5)+RR,32:GOTO 60051

65527 RESUMENEXT

65528 RETURN

65529 GOSUB380:GOSUB390:GOSUB400:GOSUB410:GOSUB42
0:GOSUB430:GOSUB435:GOSUB440:GOSUB450:GOSUB520:GO
SUB460:GOSUB470:GOSUB480:GOSUB490:GOSUB500:GOSUB51
0:RETURN
```

Listagem 2 – Montador para os blocos hexadecimais

```
10 SCREEN0:WIDTH40:SOMA=0:L=-1:C=0
20 LINE INPUT "Primeiro endereço :";E$
30 E = VAL("&H"+E$)
40 L=L+1 : IF L=16 THEN PRINT:PRINT:PRINT"Soma dos
bytes: &H"+HEX$(SOMA):L=0:TT=TT+SOMA:SOMA=0
50 PRINT : PRINT HEX$(E)+": ";
60 BYTE$=INPUT$(2)
70 R=ASC(BYTE$):IF R=13 THEN 120
80 BYTE = VAL("&H"+BYTE$)
90 POKE E,BYTE: SOMA=SOMA+BYTE : E=E+1 : PRINT
BYTE$+" ";
100 C=C+1 : IF C=8 THEN C=0 : GOTO 40
110 GOTO 60
120 PRINT:PRINT:PRINT"Soma dos bytes:
&H"+HEX$(SOMA)
130 TT=TT+SOMA : PRINT:PRINT"Soma total:
&H"+HEX$(TT)
```

LISTAGEM 3

Arquivo BAT1.ASM

Para gravar: BSAVE "BAT1.ASM",&HE000,&HE0EB

```
E000   21 00 00 06 08 CD 4A 00 4F AF CB 41 28 02 CB FF
E010   CB 49 28 02 CB F7 CB 51 28 02 CB EF CB 59 28 02
E020   CB E7 CB 61 28 02 CB DF CB 69 28 02 CB D7 CB 71
E030   28 02 CB CF CB 79 28 02 CB C7 CD 4D 00 23 10 C5
E040   C9 21 00 07 06 08 CD 4A 00 07 CD 4D 00 23 10 F6
E050   C9 01 FF 02 11 00 D0 21 00 18 CD 59 00 C9 01 FF
E060   02 11 00 18 21 00 D0 CD 5C 00 C9 AF 32 00 80 01
E070   00 40 11 01 80 21 00 80 ED B0 3E A8 D3 A8 AF 32
```

Soma dos bytes: 314C

```
E080   00 40 C7 00 F3 3A F8 F7 3D 28 17 3D 28 19 3D 28
E090   1B 3D 28 1D 3D 3D 28 1E 3D 28 20 3D 28 22 3D 28
E0A0   24 C9 CD CA E0 FB C9 CD 90 00 FB C9 CD DA E0 FB
E0B0   C9 CD 56 01 FB C9 CD 00 E0 FB C9 CD 51 E0 FB C9
E0C0   CD 5E E0 FB C9 CD 6B E0 FB C9 21 00 00 06 08 CD
E0D0   4A 00 2F CD 4D 00 23 10 F6 C9 3E A8 D3 A8 CD 00
E0E0   00 3E A0 D3 A8 C9 00 3E A0 D3 A8 C9
```

Soma dos bytes: 349D

Arquivo BAT2.ASM

Para gravar: BSAVE "BAT2.ASM",&H992B,&H9AEF

```
992B   3E A8 D3 A8 01 B0 01 11 60 79 21 3F 99 ED B0 3E
993B   A0 D3 A8 C9 00 00 00 00 01 00 03 11 00 7C 21 00
994B   18 CD 59 00 01 00 08 11 00 C0 21 00 00 CD 59 00
995B   01 20 00 11 00 C8 21 00 20 CD 59 00 11 00 60 21
996B   00 18 CD D7 07 E5 26 00 6F 29 29 29 06 08 CD D7
997B   07 23 12 13 10 F8 E1 D5 23 11 00 1B E7 D1 20 E2
998B   CD 72 00 21 00 20 11 00 7C 1A CB 3F CB 3F CB 3F
999B   E5 26 C8 6F 7E E1 06 08 CD CD 07 23 10 FA 13 E5
```

Soma dos bytes: 28F3

```
99AB   21 00 7F E7 E1 20 E2 21 00 60 01 FF 17 11 00 00
99BB   CD 5C 00 CD 00 7A CD 38 05 01 00 08 11 00 00 21
99CB   00 C0 CD 5C 00 01 20 00 11 00 20 21 00 C8 CD 5C
99DB   00 C3 F7 FF AF 32 14 7B AF 32 13 7B AF 32 12 7B
99EB   CD E1 7A 21 00 18 3A 12 7B 4F 06 00 09 3A 13 7B
99FB   87 87 87 87 87 4F 09 3A 14 7B 84 67 CD 4A 00 6F
9A0B   26 00 29 29 29 3A 14 7B 87 87 87 84 67 22 10 7B
9A1B   11 15 7B 01 08 00 CD 59 00 CD 70 7A 3A 12 7B 3C
```

Soma dos bytes: 27BF

```
9A2B   32 12 7B FE 20 20 BC CD 02 7B 3A 13 7B 3C 32 13
9A3B   7B FE 08 20 A7 3A 14 7B 3C 32 14 7B FE 03 20 98
9A4B   CD EE 7A C9 06 08 C5 CD 80 7A 3A 0C 7B CD A5 00
9A5B   C1 10 F3 C9 AF 32 0D 7B 3A 0C 7B CB 27 32 0C 7B
9A6B   3A 0D 7B 21 15 7B 06 00 4F 09 22 0E 7B 2A 10 7B
9A7B   09 01 00 20 09 CD 4A 00 E6 0F 47 3A E9 F3 B8 28
9A8B   1A CD 4A 00 CB 3F CB 3F CB 3F CB 3F 47 3A EA F3
9A9B   B8 28 10 2A 0E 7B 7E E6 80 28 08 3A 0C 7B CB C7
```

Soma dos bytes: 2FAE

```
9AAB   32 0C 7B 2A 0E 7B 7E 17 77 3A 0D 7B 3C 32 0D 7B
9ABB   FE 08 20 A4 C9 21 F5 7A 06 07 7E CD A5 00 23 10
9ACB   F9 C9 21 FC 7A 06 06 18 F1 1B 41 08 1B 4B 00 01
9ADB   1B 41 0D 1B 51 27 3E 0D CD A5 00 3E 0A C3 A5 00
9AEB   00 B7 ED 52 38
```

Soma dos bytes: 1653

Arquivo BAT3.ASM

Para gravar: BSAVE "BAT3.ASM",&H9100,&H9829,&H9800

```
9100   00 00 00 00 00 00 00 00 38 28 28 28 38 00 38 00
9110   6C 6C 6C 00 00 00 00 00 6C FE FE 6C FE FE 6C 00
9120   30 7E B0 7C 32 FC 30 00 C4 CC 18 30 60 CC 8C 00
9130   30 68 30 6A A4 AA 70 00 18 28 50 60 00 00 00 00
9140   38 50 A0 A0 A0 50 38 00 E0 50 28 28 28 50 E0 00
9150   20 A8 70 20 70 A8 20 00 00 30 30 FC FC 30 30 00
9160   00 00 00 00 18 18 18 30 00 00 7E 42 7E 00 00 00
9170   00 00 00 00 70 50 70 00 00 00 18 28 50 A0 C0 00
```

Soma dos bytes: 222E

```
9180   7C A2 A2 A2 A2 A2 7C 00 38 68 28 28 28 28 7C 00
9190   7C 8A 0A 7C A0 A2 FC 00 7C 8A 8A 3C 0A 8A 7C 00
91A0   1C 34 54 94 F6 14 1C 00 FE A0 A0 FC 02 82 7C 00
91B0   7C A2 A0 BC A2 A2 7C 00 FE 0A 14 28 28 28 38 00
91C0   7C A2 A2 7C A2 A2 7C 00 7C 8A 8A 7A 0A 8A 7C 00
91D0   00 30 30 00 00 30 30 00 00 18 18 00 18 18 18 30
91E0   1C 28 50 A0 50 28 1C 00 00 F8 F8 00 F8 F8 00 00
91F0   C0 A0 50 28 50 A0 C0 00 7C 8A 8A 1C 30 00 30 00
```

Soma dos bytes: 2CDC

```
9200   7C 8A 0A 4A AA AA 7C 00 7C A2 A2 BE A2 A2 E2 00
9210   FC A2 A2 BC A2 A2 FC 00 7C A2 A0 A0 A0 A2 7C 00
9220   FC A2 A2 A2 A2 A2 FC 00 FE A0 A0 BC A0 A0 FE 00
9230   FE A0 A0 BC A0 A0 E0 00 7C A2 A0 AE A2 A2 7C 00
9240   E2 A2 A2 BE A2 A2 E2 00 7C 28 28 28 28 28 7C 00
9250   3E 14 14 14 94 94 78 00 E2 A4 A8 B4 A2 A2 E2 00
9260   E0 A0 A0 A0 A0 A0 FE 00 E2 B6 AA A2 A2 A2 E2 00
9270   E2 B2 B2 AA A6 A6 E2 00 FE A2 A2 A2 A2 A2 FE 00
```

Soma dos bytes: 46FE

```
9280  FC A2 A2 BC A0 A0 E0 00 7C A2 A2 A2 AA A4 7A 00
9290  FC A2 A2 BC A8 A4 E2 00 7C A2 A0 7C 0A 8A 7C 00
92A0  FE 28 28 28 28 28 38 00 E2 A2 A2 A2 A2 A2 7C 00
92B0  E2 A2 A2 A2 64 28 10 00 82 92 92 AA D6 AA C6 00
92C0  C2 A4 58 28 34 4A 86 00 E2 A2 A2 7C 10 10 10 00
92D0  FE 0A 14 28 50 A0 FE 00 78 50 50 50 50 50 78 00
92E0  00 00 60 50 28 14 0C 00 78 28 28 28 28 28 78 00
92F0  20 50 A8 D8 00 00 00 00 00 00 00 00 F8 88 F8 00
```

Soma dos bytes: 32C0

```
9300  00 00 7C A2 A0 A2 7C 10 00 00 7C 0A 7A 8A 7C 00
9310  E0 A0 BC A2 A2 A2 FC 00 00 00 7C A2 A0 A2 7C 00
9320  0E 0A 7A 8A 8A 8A 7E 00 00 00 7C A2 BE A0 7E 00
9330  3C 52 50 DC 50 50 70 00 00 00 7E 8A 8A 7A 0A 7C
9340  E0 A0 BC A2 A2 A2 E2 00 38 00 78 28 28 28 7C 00
9350  1C 00 3E 14 14 14 94 78 70 52 54 58 58 54 72 00
9360  78 28 28 28 28 28 7C 00 00 00 F4 AA AA AA EA 00
9370  00 00 EC B2 A2 A2 E2 00 00 00 7C A2 A2 A2 7C 00
```

Soma dos bytes: 2F00

```
9380  00 00 FC A2 A2 BC A0 E0 00 00 7E 8A 9A 6A 0A 0E
9390  00 00 EC B2 A0 A0 E0 00 00 00 7E A0 7C 0A FC 00
93A0  70 50 D8 50 50 52 3C 00 00 00 E4 A4 A4 A4 7A 00
93B0  00 00 E2 A2 A2 64 38 00 00 00 82 92 D6 BA 6C 00
93C0  00 00 E6 58 28 34 CE 00 00 00 8E 8A 7A 0A 8A 7C
93D0  00 00 FE 14 28 50 FE 00 1C 28 28 70 28 28 1C 00
93E0  30 30 30 00 30 30 30 00 E0 50 50 38 50 50 E0 00
93F0  7C A2 A0 A0 A0 A2 7C 20 00 00 20 50 F8 00 00 00
```

Soma dos bytes: 2BB6

```
9400  40 E8 B8 10 00 00 00 00 50 A0 00 90 90 90 68 00
9410  08 10 70 88 F8 80 70 00 20 50 70 08 78 88 78 00
9420  50 A0 70 08 78 88 78 00 80 40 70 08 78 88 78 00
9430  20 00 70 08 78 88 78 00 80 40 20 00 00 00 00 00
9440  20 50 70 88 F8 80 70 00 50 A0 70 88 F8 80 70 00
9450  80 40 70 88 F8 80 70 00 50 A0 00 60 20 20 70 00
9460  20 50 00 60 20 20 70 00 80 40 00 60 20 20 70 00
9470  50 A0 50 88 F8 88 88 00 20 20 50 88 F8 88 88 00
```

Soma dos bytes: 29D0

```
9480  38 68 28 28 28 28 7C 00 7C 8A 0A 7C A0 A2 FC 00
9490  7C 8A 8A 3C 0A 8A 7C 00 1C 34 54 94 F6 14 1C 00
94A0  FE A0 A0 FC 02 82 7C 00 7C A2 A0 BC A2 A2 7C 00
94B0  FE 0A 14 28 28 28 38 00 7C A2 A2 7C A2 A2 7C 00
94C0  7C 8A 8A 7A 0A 8A 7C 00 20 A8 70 20 70 A8 20 00
94D0  00 A0 88 88 88 88 70 00 20 20 78 80 80 78 20 20
94E0  18 24 20 F8 20 E2 5C 00 88 50 20 F8 20 F8 20 00
94F0  40 A0 80 8A 8C A8 48 00 18 20 20 F8 20 20 20 40
```

Soma dos bytes: 2F76

```
9500  10 20 70 08 78 88 78 00 10 20 00 60 20 20 70 00
9510  08 10 70 88 88 88 70 00 10 20 90 90 90 90 68 00
9520  50 A0 00 A0 D0 90 90 00 28 50 00 C8 A8 98 88 00
9530  60 90 90 68 00 F8 00 00 60 90 90 60 00 F0 00 00
9540  20 00 20 40 80 88 70 00 00 00 00 F8 80 80 00 00
9550  FF 80 80 80 FF 10 10 10 00 00 00 00 00 00 00 00
9560  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
9570  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Soma dos bytes: 1C9E

```
9580  00 00 00 00 00 00 00 00 28 50 70 08 78 88 78 00
9590  28 50 00 70 20 20 70 00 28 50 00 20 20 20 70 00
95A0  50 A0 70 88 88 88 70 00 50 A0 70 88 88 88 70 00
95B0  50 A0 88 88 88 88 70 00 50 A0 00 90 90 90 68 00
95C0  FC 48 48 48 E8 08 50 20 00 A0 00 A0 A0 A0 20 40
95D0  C0 44 C8 54 EC 54 9E 04 07 1F 3F 7F 7F FF FF FF
95E0  E0 F8 FC FE FE FF FF FF 00 00 00 00 00 00 00 00
95F0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Soma dos bytes: 28E3

```
9600  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
9610  00 00 FF FF FF FF FF FF FF FF 00 00 00 00 00 00
9620  00 00 00 3C 3C 00 00 00 FF FF FF FF FF FF 00 00
9630  C0 C0 C0 C0 C0 C0 C0 C0 0F 0F 0F 0F F0 F0 F0 F0
9640  FC FC FC FC FC FC FC FC 03 03 03 03 03 03 03 03
9650  3F 3F 3F 3F 3F 3F 3F 3F 11 22 44 88 11 22 44 88
9660  FF FF FF 7F 7F 3F 1F 07 FF FF FF FE FE FC F8 E0
9670  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Soma dos bytes: 3081

```
9680  3A 3A 12 FE 3A 38 44 82 00 00 00 00 00 00 00 00
9690  00 00 00 00 00 00 00 00 F0 F0 F0 F0 00 00 00 00
96A0  00 00 00 00 0F 0F 0F 0F 0F 0F 0F 0F 00 00 00 00
96B0  00 00 00 00 F0 F0 F0 F0 33 33 CC CC 33 33 CC CC
96C0  00 10 38 54 BA 54 92 10 20 20 70 20 70 20 20 00
96D0  00 00 00 50 88 A8 50 00 EE AA 8A EC 2A AA EA 00
96E0  00 00 00 00 FF FF FF FF 07 04 08 10 30 40 40 80
96F0  C0 2C 1A 01 00 00 00 00 01 36 48 80 00 00 00 00
```

Soma dos bytes: 206D

```
9700  FF 9F 66 F9 FF 9F 66 F9 00 00 00 00 00 00 00 00
9710  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
9720  F8 80 40 20 40 88 F8 00 00 00 78 90 90 90 60 00
9730  00 50 50 50 50 68 80 80 00 50 A0 20 20 20 20 00
9740  02 5C 22 66 22 94 4F 30 00 10 38 7C 38 10 00 00
9750  70 88 88 88 50 50 D8 00 30 40 40 20 50 50 50 20
9760  00 00 00 50 A8 A8 50 00 08 70 A8 A8 A8 70 80 00
9770  80 80 80 40 30 0E 01 00 00 00 00 00 03 1C E0 00
```

Soma dos bytes: 210F

```
9780  AA FE FE AA 92 BA 44 38 00 F8 FE F8 FE FE BA EE
9790  60 68 52 66 FA F8 88 88 3A 3A 12 FE 3A 38 44 82
97A0  04 0E 1C 3C 62 EA E2 FC 00 F8 FE F8 FE FE BA EE
97B0  60 68 52 66 FA F8 88 88 3A 3A 12 FE 3A 38 44 82
97C0  AA FE FE AA 92 BA 44 38 00 F8 FE F8 FE FE BA EE
97D0  60 68 52 66 FA F8 88 88 3A 3A 12 FE 3A 38 44 82
97E0  04 0E 1C 3C 62 EA E2 FC 00 F8 FE F8 FE FE BA EE
97F0  60 68 52 66 FA F8 88 88 FF FF FF FF FF FF FF FF
```

Soma dos bytes: 4DD4

```
9800  CD 6F 00 3E 1D 32 AF F3 32 B0 F3 3E 0F 32 E9 F3
9810  3E 0B 32 EA F3 3E 01 32 EB F3 CD 62 00 01 FF 07
9820  11 00 00 21 00 90 CD 5C 00 C9
```

Soma dos bytes: 112C

Arquivo BAT4.ASM

Para gravar: BSAVE "BAT4.ASM",&H984F,&H98BE,&H9875

```
984F  CB 1B CB CB CB CB CB CB CB CB CB CB CB CB CB CB
985F  CB CB B1 B1 CB CB CB CB CB CB 1B CB B4 1D 0B 1B
986F  00 00 00 00 00 00 01 20 00 21 4F 98 11 00 20 CD
987F  5C 00 11 00 00 06 17 EB 11 20 00 ED 5A EB 21 00
988F  18 ED 5A 3E DB CD 4D 00 2B CD 4D 00 10 E9 16 00
989F  06 20 21 00 18 58 ED 5A 3E DB CD 4D 00 11 E0 02
98AF  ED 5A CD 4D 00 16 00 10 E9 21 00 18 CD 4D 00 C9
```

Soma dos bytes: 2CCB

Acelerando o Turbo Pascal

MSX BITS

Fernando da Rocha Carneiro

Deixo aqui um pequeno "macete" para o pessoal que programa em Turbo Pascal no MSX. Trata-se de uma pequena mudança na rotina Write (a encarregada de jogar os dados no vídeo) para que a mesma funcione com uma velocidade aproximadamente quatro vezes maior.

O que fiz foi mudar o endereço que a rotina Write/Writeln usa para jogar os dados no vídeo pela CHPUT da BIOS, a mesma utilizada pelo BASIC para desempenhar a mesma função (o comando Print).

Com isso temos a rotina padrão Write direcionada para "Print", que se encarrega de passar o caracter a ser impresso para o registrador A e chama a rotina CHPUT do BIOS. A rotina padrão permanece inalterada, mantendo, portanto, as poderosas funções das procedures Write/Writeln.

Devemos apenas mudar a variável do sistema ConOutPtr para o endereço da rotina logo no início do seu programa (preferencialmente) e pronto.

Lembre-se que, assim como a função 2 do BDOS, CHPUT emula o padrão VT-52, só que muito mais rápido. A única desvantagem é que CHPUT não faz a verificação das teclas CONTROL+S e CONTROL+C, mas isto não é problema pois elas podem ser verificadas pelo programa em Pascal sem comprometer a velocidade de impressão no vídeo. ❑

LISTAGEM 1

```
Procedure Print(x:char);
Begin
  Inline($3A/x/   { LD A,(X) ; Carrega o
                             ; Reg. A com
                             ; conteúdo da
                             ; variável X }
         $F7/     { RST 30h  ; "CALLF"     }
         $00/     { DEFB 00  ; Slot 0      }
         $A2/00);{ DEFW 0A2 ; Endereço
                             ;00A2h (CHPUT)}
end;

Begin      { Bloco principal do seu programa }
  ConOutPtr:=Addr(Print);
   { Desvia rotina padrão Write para PRINT }
      .
      .
      .
end.     { Fim do seu programa }
```

JOGO

Magnar

A nova safra de jogos europeus

José Agripino

- Parallax 92
- MSX2
- Mapper
- FM

Após um longo jejum de softs europeus para MSX a Parallax colocou no mercado o primeiro de um série de jogos que irão fazê-lo abandonar o dicionário de japonês.

Certo dia, ao fazer um levantamento dos jogos para MSX2, constatei que, ao contrário da crença geral, existem mais jogos de ação do que RPGs para esta linha de micros. É bem verdade que por RPG eu entendo jogos como Dragon Slayer 6 e SD Snatcher, onde os combate são realizados via menus. Jogos como Xak ou Y's devem ser classificados como RPG-Ação pois os combates são resolvidos com o bom e velho joystick.

E quanto aos jogos de ação pura? Muitos deles são jogos de espaçonave como Aleste e Hydefos. Restam jogos como Vampire Killer, Psycho World, FireHawk etc., todos oriundos da Terra do Sol Nascente.

Mas será que a velha Europa esqueceu do MSX? Bem, o MSX foi totalmente ignorado na Inglaterra, sede de grandes empresas de software de lazer. Porém em países como a Holanda, a Bélgica, França, Suíça e Espanha o público permaneceu fiel ao MSX. Agora, com a já iniciada invasão dos MSX R na Europa, certas empresas começaram a demonstrar um novo interesse pelo padrão. Para demonstrar esta tendência, temos jogos como Terminator II, Dizzy e outros que brevemente estarão no mercado de MSX 2, 2+ e Turbo R.

E o que isso traz de bom para os usuários de software de lazer? Primeiro, obviamente, a facilidade de língua: nada mais de japonês! Segundo, o usuário de MSX deve sentir um certa inveja de jogos de Amiga como Loom, Lure of Temptress, Another World, Shadow of the Beast etc., jogos que ultrapassam o rótulo de excelente. Não me entendam mal, isso não quer dizer que tais jogos serão convertidos para o MSX. Quero apenas mostrar que estes jogos são um exemplo do que as softhouses ocidentais produzem, em comparação com o estilo japonês que valoriza a tríade RPG-Estratégia-Puzzle.

Em Magnar este estilo ocidental é notado logo na tela-título do jogo, embalada por uma trilha sonora tipicamente européia (apenas no FM). Em seguida, o demo do jogo, contando uma estória completamente sem nexo, mas muito bem feita. O menu de início de jogo permite a escolha do sistema NTSC/PAL-M.

O JOGO

Magnar consiste de oito missões onde o jogador controla um tanque entre as paredes, complexos industriais, prisões e outras localidades menos agradáveis. As missões estão distribuídas em três discos e o símbolo para a troca dos mesmos é um disco 3 1/2 numerado. Bem europeu.

Um ponto interessante do jogo é que ele pode ser jogado tanto pelo teclado como pelo joystick padrão, com igual performance, já que é possível acessar o menu de armas pressionando os botões A + B do controle.

Em termos de gráficos, o jogo é agradável, pois não utiliza o scroll, com sua estrutura sendo formada por salas. Isto deu mais liberdade para a criação dos detalhes no cenário. O som, para quem dispõe de um cartucho FM, desaponta por não ser aquilo que se espera após ouvir a excelente introdução. Sem o FM, apenas os ruídos de tiro são ouvidos.

Pelo fato de utilizar a Mapper o jogo não acessa o disco durante a fase, tornando o jogo mais agradável. Não é um jogo difícil, excetuando-se uma fase com paredes invisíveis. Resumindo, é um excelente descanso mental após um bom RPG e pode ser concluído em poucas horas.

Em tempo: a Electronic Arts uniu-se à JVC no Japão para a criação de softs mais famosos do país. Com isso, quem sabe não tenhamos um "NBA Lakers x Celtics" para o MSX em breve? ❑

Jogo

Maze of Galious

Knightmare II

José L. M. do Amaral Jr.

Dentre os melhores jogos para MSX, destaca-se a série KNIGHTMARE. No primeiro (sem MEGARAM), nosso herói Popolon tem que salvar sua amada, Aphrodite. No segundo (com MEGARAM), o casal, já unido, deve resgatar seu filho. A seguir, conheceremos o jogo e as dicas para concluí-lo.

O JOGO

O Castelo (Castle)

Apesar da primeira impressão, o castelo não é um labirinto muito complexo. Nele encontram-se alguns itens que facilitam nossa missão. Eles encontram-se escondidos em bolas vermelhas, adversários e cavernas, as quais muitas vezes também estão ocultas nas referidas bolas. Dentre as cavernas, destaco a que nos fornece o código para continuarmos o jogo em outra oportunidade. Ela encontra-se na quarta sala contando a partir da em que o jogo inicia-se. Pode-se alcançá-la mais rapidamente teclando F1 seguido de RETURN, desde que possua-se o anel (item).

Usando a tecla F1 também selecionamos o Popolon (golpes mais fortes e saltos maiores) e Aphrodite (resiste mais na água). Mais embaixo, podemos selecionar as armas colocando o joystick para baixo. Ao todo são cinco armas e uma lupa, ativadas pelo botão superior ou pela tecla "M". O botão inferior ou o espaço ativam a espada. Voltando às armas:

A) flecha vermelha: mais rápida;

B) flecha branca: mais destrutiva porém lenta;

C) foguinho: desce sempre em frente;

D) estrelinha: circula plataformas;

E) mina: arma mais poderosa – explode ao ser pisada;

F) lupa: mostra o nome do guardião impresso em pedras monumento, encontradas nos mundos.

Dentre os itens, destaco a "cruz inclinada" (ela é mostrada inclinada) que aumenta a energia do personagem. Há uma na bola vermelha superior da sala, no lado direito da sala em que inicia-se o jogo. Obtemos outras destruindo um guardião. A energia é recomposta quando é preenchido o marcador mais embaixo. Isto é feito matando-se adversários.

Os mundos (Worlds)

Os mundos às vezes se fazem como fases neste jogo. Entramos neles pelas portas vermelhas do castelo. É importante observar que devemos acessá-los em ordem: WORLD 1, WORLD 2 etc. Sabemos em qual entrar através do toque de um sino (item). Cada um abriga um guardião que para poder ser destruído, requer que possuamos cinco objetos:

A) o mapa do respectivo mundo;

B) um cetro;

C) uma garrafa com "água pura";

D) um objeto vermelho que parece uma galinha assada;

E) o nome do guardião: encontrado em uma das "pedras monumento" de um mundo, deve ser lido com a lupa e digitado na sala azul do mapa, onde há um grande símbolo. Digitado o nome, o monstro surge, bastando destruí-lo para concluir a fase (ao todo são dez).

DICAS

Antes de ir aos mundos, percorra o castelo em busca dos itens, em especial a "cruz inclinada", o anel, o sino e a harpa (basta um golpe de espada para destruir as bolas vermelhas de uma sala). Busque ainda as armas (umas estão em cavernas e outras em bolas vermelhas), a lupa (em uma caverna escondida por uma bola vermelha), munição, chaves e moedas. Para inserir o código de uma jogada anterior, tecle CONTROL+L na tela que pede para pressionar espaço. Para os mais apressados, seguem os nomes dos guardiões (entre parênteses) e as dicas sobre cada mundo:

World 1 (Yomar)

Esqueleto de dragão. Deve ser eliminado com as flechas vermelhas. As aranhas correm em sua direção quando coloca-se o personagem na mesma horizontal delas.

World 2 (Elohim)

Planta que é destruída usando flechas brancas. Antes da sala do guardião, há um poço aparentemente intransponível. Para transpô-lo, pegue o garoto escondido na parede do lado direito de uma das pedras monumento do mundo. A parede deverá ser golpeada com a espada, mas antes, o fantasma deve ser morto. Note que ao golpear a parede, você ouvirá um som característico.

World 3 (Hahakla)

É um dragão que quando surgir numa plataforma, basta subir para a plataforma acima e disparar o foguinho contra ele várias vezes.

World 4 (Barechet)

É uma "bolha" que você destruirá usando minas. Antes disso destrua todos os morcegos que encontrar e desça a escada sem saída (ache-a e descubra a razão golpeando suas paredes com a espada).

World 5 (Heotymeo)

Dragão vulnerável à flecha vermelha. Antes do guardião, você voltará a deparar-se com um lago que parece intransponível. Atire flechas vermelhas para a outra margem até

surgir uma pedra. Continue fazendo isso até atravessar o poço. Dispare as flechas da beirada da margem na passagem inferior.

World 6 (Lepha)
Caranguejo que você poderá vencer usando o foguinho. Mate todos os adversários e siga a instrução do DEMETER.

World 7 (Nawabra)
É uma boca que morre ao ser atingida por flechas brancas. Mergulhe no lago sem piso no fundo para alcançar outras salas.

World 8 (Ascher)
Outro Caranguejo, só que este é mais vulnerável às estrelinhas. Você deverá dispará-las da plataforma acima daquela na qual o monstro está ou da mesma. Nesta fase, o cetro é dispensável.

World 9 (Xywoleh)
Dragão de três cabeças que você destruirá utilizando flechas brancas. Destrua os adversários da segunda sala para que surja uma escada. Também destrua um monstro azul da sala ao lado da do guardião. Ao destruí-lo, note que um bloco do teto desaparecerá. Pule e golpeie com a espada a parede à esquerda do espaço que surgiu (lembre-se do som característico). Você fará surgir a pedra monumento com o nome do guardião e a passagem para a sala do mesmo.

World 10 (Hamalech)
A entrada deste mundo pode aparecer em qualquer lugar do castelo. Assim sendo, ao concluir o mundo 9, anote o código na caverna do DEMETER, recarregue o jogo e digite-o. Há boas chances do mundo 10 surgir na sala de início (não teremos que procurá-lo). Caso você não consiga na primeira vez, repita a operação até o portal surgir na sala inicial. O mundo 10 tem cinco salas em linha, sendo que você alcançará uma outra no extremo direito do mundo, ao matar o guardião, onde estará o bebê.

Ataque o monstro inicialmente com flechas brancas e, após destruir as bolas que ficam girando, use as vermelhas. Vá para a sala final e descubra em qual das bolas vermelhas está o bebê.

Observações
As armas que citei para superar cada guardião são as mais eficazes em minha opinião, podendo ser usadas outras em alguns casos. Há paredes que só somem ou escadas que só surgem se você destruir um certo adversário ou um grupo deles em certa sala. Assim, aconselho eliminar todos os adversários de um mundo. Existem momentos nos quais a Aphrodite é mais eficiente por ela poder disparar mais tiros.

UM ITEM DIFÍCIL

Além da "cruz inclinada", há uma outra cruz, reta, muito difícil de ser descoberta, já que encontra-se em uma sala secreta do castelo. Para alcançá-la, vá até a sala onde está a caverna dos códigos (DEMETER), vá para a sala da direita, suba até a próxima (onde há dois soldados azuis), desça de volta pelo outro lado, avance duas salas para a direita e torne a subir para a próxima sala. Nesta, deverão existir borboletas e a entrada para o mundo 5.

Prosseguindo, volte para a sala anterior pelo outro lado, avance mais duas salas para a direita e retorne uma para a esquerda (por baixo), indo a seguir para a sala debaixo. Esta sala deverá ter alguns porcos-espinho. Por fim, usando a Aphrodite, desça mais uma sala.

Se você seguiu todos os passos corretamente, a sala que você está é um "poço" mais ou menos estreito com duas pequenas plataformas. Posicione-se na ponta esquerda da plataforma debaixo. Salte e golpeie a parede esquerda, retornando para a plataforma (lembre-se do som característico). Repita a operação mais duas vezes. Caso você tenha acertado os golpes, surgirá um espaço aparentemente sem saída com uma moeda. Alcance-o e pule contra a parede até conseguir penetrar na mesma. Ao conseguir, caminhe com calma para a esquerda até cair e pare. Agora pule mais uma vez direcionando-se para a esquerda (você ainda deverá estar na sala do poço, mas "dentro" da parede).

Finalmente basta caminhar para a esquerda até a sala secreta que esconde a "cruz reta" em uma bola vermelha. Tente descobrir sua utilidade.

DOIS CÓDIGOS

Seguem-se dois códigos. O primeiro lhe dará a maioria dos itens (não inclui-se a "cruz reta") e todas as armas, mas nenhum dos mundos resolvidos. O segundo traz todos os itens que consegui, faltando apenas o mundo 10 a ser encontrado e resolvido.

Maioria dos itens e armas
KORM 6RR6 UR4F 4237
UR5F UR3F UR3F UR3F
UYTP NMWS O8S6 F

Faltando apenas o Mundo 10
NGRA 679M UR4F 123N
ULWJ C9SP WLWJ J9SX
2YTI CMWI OLO6 9

CONCLUSÃO

Pelo grafismo e complexidade, este é um jogo que devemos concluir. Em caso de dúvidas, escreva-me e terei prazer em ajudá-lo.

José Amaral Jr.
Rua Santa Clara, 84
Partenon – Porto Alegre – RS
Cep 90680-540
Tel.: (051) 336-6909

Colaboração: Cristian Piva Franzen.

Caros amigos de CPU,

Eu era usuário de um MSX 2, mas estava insatisfeito pois apesar do potencial da máquina, não havia bons softwares aplicativos disponíveis no mercado (na época eles estavam apenas começando a surgir). Por adorar programação e querer extrair toda a capacidade que a linha 2.0 possui, comecei a estudar Assembly, mas não encontrei dados sobre as implementações do VDP, ROM etc. Então cheguei à seguinte conclusão: gastei dinheiro à toa com a transformação! Não queria um MSX 2 apenas para jogos e editores gráficos, entendem? Acreditava (e ainda acredito) no potencial do micro, mas não havia informação técnica disponível, tampouco atitudes sérias por parte dos programadores brasileiros (que já tinham mostrado do que eram capazes de fazer com os MSX 1) quanto à capacidade de seu irmão mais novo.

Há dois anos possuo um Amiga 500 e confesso que estou muito satisfeito com ele, apesar de toda sua literatura ser importada e estar em inglês. Com o surgimento do Caderno Amiga na revista, voltei a comprá-la com alguma regularidade e quando, inocentemente (e com um certo ar irônico, admito), resolvi dar uma olhada no Caderno MSX para ver o que ainda havia a ser dito sobre o MSX e... Surpresa! Não é que o MSX não só se mantém corajosamente de pé, como agora também já se pode obter facilmente todo o tipo de informação que eu procurei incansavelmente tempos atrás? Levado então pelo entusiasmo, como quem houvesse conhecido o micrinho naquele exato momento, tirei-o do armário, espanei a poeira, peguei de volta os disquetes emprestados com os colegas e resolvi pôr o velho e simpático MSX de volta à ativa!

O que eu pretendo mostrar com toda essa história é que o usuário do MSX não tem que se preocupar com a existência do novo espaço do Amiga na revista, pois quem gosta da máquina, com o devido suporte, pode comprar qualquer outro modelo de micro, mas terá sempre o pequenino do lado esquerdo da mesa. E aqueles que dizem que o MSX está morto, que não serve mais para nada ou que a era dos 8 bits já passou, é porque não sentem prazer ao sentar-se em frente a um MSX e certamente nunca souberam o que esse micro tem a oferecer. Sim, certamente não sabem!

Ah! Gostaria de pedir aos usuários do MSX, principalmente do MSX 2, que programem "fluentemente" em Assembly, para entrarem em contato comigo.

Omar Silas Colocci
Rua Maruim, 501/303
Aracaju – SE
Tel.: (079) 222-1956
Endereço eletrônico:
ALUNOINF@BRUFSE.BITNET

Prezado Omar,

Me sinto particularmente feliz com manifestações espontâneas como a sua. O seu depoimento, entretanto, tem um peso ainda maior por você também ser usuário de outra linha. O MSX realmente tem o poder de cativar e certamente não cairá em desuso tão cedo, como apregoa-se por aí. Ainda há muito o que se fazer com esse micro, qualquer que seja a versão.

É esse o compromisso de CPU: instigar os leitores revelando novas facetas deste micro fantástico. Não é uma tarefa fácil, mas temos alguns pontos a nosso favor. Os sete anos de estrada do MSX no Brasil serviram para "formar" várias levas de usuários. Hoje o MSX é um padrão amadurecido no país e podemos presenciar atitudes sérias de várias empresas, como a ACVS e Tecnobytes em hardware e muitas outras em software. Mas nada disso seria possível se não houvesse um grande interesse dos usuários.

Apenas uma curiosidade: Por quê colocar o MSX no "lado esquerdo da mesa"? Você é canhoto?

Carlos Alberto Herszterg

• • •

Prezados amigos da CPU/MSX,

Sou leitor assíduo da revista desde que a conheci há cerca de um ano, quando adquiri meu MSX. Aprecio muito a revista e leio todos os artigos, mesmo os que se referem ao Amiga, apesar de só a título de curiosidade, pois não pretendo trocar meu tão amado e útil MSX. Possuo um Expert DD Plus, com drives 3 1/4 e 5 1/2 e impressora Olívia. Estou satisfeitíssimo com meu equipamento, que não tem apresentado nenhum problema desde que o adquiri. Pretendo expandi-lo em breve com o cartão MSX 2+ e com uma Megaram. A revista tem me ajudado muito a conhecer e esclarecer dúvidas e principalmente a conhecer fornecedores e gente interessada em preservar o MSX. E é exatamente isso que me levou a escrever.

Aqui em Belo Horizonte temos que batalhar para adquirir software e hardware para o MSX. As poucas lojas estabelecidas que vendem programas (só conheço duas), vendem apenas programas piratas, que além de não funcionarem direito, não vêm com o suporte técnico nem manuais. Além disso, temos de enfrentar um péssimo atendimento e um tratamento de subserviço, já que nessas lojas predominam os PCs e amigas da vida, ficando o MSX e seus usuários esquecidos em um canto qualquer. Lamento profundamente que isso ocorra, pois acredito que ainda haja gente aqui em BH, que, como eu, pretende usar programas de qualidade e confiabilidade em seu MSX.

Gostaria de saber o que eu poderia fazer para ser revendedor de produtos originais para MSX, tanto programas como periféricos, já que nenhuma das lojas que conheço aqui faz isso, pelo menos de maneira adequada. Tenho disposição e um tremendo interesse por isso. Gostaria que me orientassem nesse sentido, dando uma força a todos nós usuários do MSX em BH.

Sei que a maior parte dos usuários se prende mesmo aos jogos, o que os piratas daqui e de outros estados servem bem. Mas para mim e outras pessoas, o MSX serve para muito mais que isso, apesar da maioria dos usuários desconhecer ou ignorar. Sou universitário e utilizo o MSX em todos os meus trabalhos escolares, além de usar planilhas para balanço em uma microempresa de minha família. Tenho interesse ainda em me desenvolver na área gráfica, mas ainda não adquiri os programas necessários.

Gostaria ainda que me esclarecessem uma dúvida. Qual a diferença entre a Memory Mapper e a Megaram? Qual me seria mais útil para aumentar a memória para processamento de textos? Gostaria também de saber como faço para adquirir os números atrasados da revista.

Alex Assírio Bossi, Belo Horizonte

Caro Alex,

É isso aí, você como muitas outras pessoas (eu também!) descobriram que o MSX pode perfeitamente dar conta do recado, seja qual for a área do nosso interesse. Por isso é difícil compreender as razões das pessoas que "migram" para outras linhas, desembolsando MUITAS centenas de dólares, apenas para usar joguinhos. Para a grande maioria destes, um MSX 2+ seria ideal – a transformação realmente vale a pena.

*Fico satisfeito de ver gente interessada em trabalhar com o MSX, ajudando a manter o padrão. Para revender softwares e periféricos do MSX basta que você entre em contato com os legítimos detentores dos direitos dos produtos que você deseja comercializar e fazer-lhes uma **proposta de representação**. A coisa funciona assim: você é cadastrado por essas empresas como um **representante autorizado** (com contrato em alguns casos) e assim tem*

CARTAS

direito a um determinado percentual, que lhe é concedido sob a forma de desconto, na compra de um número mínimo de cópias de algum produto. Com esse acordo, a produtora responsabiliza-se pelo suporte aos produtos e assim você pode repassar esse suporte para seus clientes. Certamente todos os anunciantes de CPU concedem representações, desde que a proposta seja séria.

Quanto a sua dúvida, se você chegou até aqui (cartas), já leu o artigo da Memory Mapper, que também desvenda algumas diferenças entre os dois padrões de memória. Em todo caso, nenhuma dessas expansões por si só aumentam o espaço para a edição de textos. É preciso que o software reconheça esta memória adicional.

As edições atrasadas de CPU podem ser obtidas mediante uma carta à Bônus Rio Editora Ltda., acompanhada de cheque nominal ou vale postal (agência Copacabana) no valor do preço de capa atual para cada número em atraso.

Carlos Alberto Herszteg

• • •

Venho acompanhando CPU há algum tempo e, por não possuir todos os volumes, tenho as seguintes perguntas:
1) Como posso conseguir o programa MSXDEBUG juntamente com manual do mesmo?
2) O esquema da MEGARAM foi publicado conforme o anúncio em CPU 26 página 66? Caso já tenha sido publicado, em qual edição?
3) Gostaria de ver publicado algum artigo sobre os pacotes de assembler no MSX (eu faço uso de DEVPAC mas desconheço os outros).
4) Existe alguma versão do DEVPAC para o TURBO R?
5) Gostaria de alguma dica para o VENON STRIKES BACK.

Peço que publiquem meu endereço e, caso algum leitor tenha resposta para alguma dúvida minha, gostaria que me escrevesse. Tenho um Expert 1.0 com megaram e drive de 5 1/4.

Para terminar, gostaria, como mais um MSX-maníaco, parabenizar os senhores editores por ainda trabalhar com o MSX e pela qualidade dos artigos publicados.

Wagner Fernandes Silva
Rua Defonso Albano, 441/303
Fortaleza - CE
Cep 60165-000

Caro Wagner,

Na ordem de suas perguntas:
1) O MSXDEBUG, um programa sob a forma de projeto de Sérgio Duric Calheiros, esteve presente em CPU/MSX nas edições de número 7 até 20. Nestas edições, cada novo comando implementado foi acompanhado das explicações necessárias, além das instruções de uso. Na edição 21 foi publicado o programa na íntegra e em CPU 26, os últimos comandos que fizeram parte da montagem "oficial".
2) O esquema da Megaram esteve por ser publicado, mas devido a alguns "atropelos" acabou sendo adiado. Compensamos esta falta com a Memory Mapper publicada nesta edição. De qualquer forma, aguarde as novidades!
3) Pacotes de qualidade para programação Assembly existem apenas dois na minha opinião: um é o Devpac e o outro é o Macro-80. O problema das abordagens desses programas é que elas seriam grandes demais. Não adianta falar por alto: o que estamos tentando viabilizar é a publicação de guias completos de utilização.
4) Apesar de não conhecer nenhum, deve existir algum assemblador para o R800.
5) Aí vai a dica que você pediu. Para entrar com as senhas, tecle "P". As senhas são: MAYHEN, TRANSMOGRIFY, VALKYR e PETALS OF DOOM.

Valeu Wagner, a equipe agradece os elogios.

Carlos Alberto Herszterg

• • •

Pude reparar que, de umas publicações para cá, o conteúdo da CPU tem se tornado cada vez mais interessante. E, pensando na continuidade do nível de suas matérias, gostaria que fossem publicadas informações sobre o uso de gráficos no Turbo Pascal.

Gostaria também que saísse nessa revista uma lista dos principais aplicativos gráficos para o MSX 2 e 2+, já que pretendo transformar brevemente o meu MSX 1 para MSX 2+.

Guilherme S. Pereira - Vila Velha, ES

Caro Guilherme,

Pude reparar que, desde a CPU 32 (a rigor desde CPU 30, modéstia a parte), os leitores responderam positivamente às mudanças que fizemos na revista. Apesar de estarmos com o mesmo espaço "físico", o espaço "lógico" dobrou da edição 32 em diante. Sem alarde, aumentamos muito o número de matérias e conseguimos colocar os assuntos de real interesse dos leitores em seções fixas da revista, como o MSX 2, dicas de jogos, Assembler, além dos "MSX Bits", uma idéia que eu vinha amadurecendo há algum tempo.

Os recursos gráficos no Turbo Pascal já foram tratados em CPU/MSX 10, 15 e 16, mas nada impede que os leitores "feras" nessa linguagem nos enviem artigos sobre novos comandos gráficos ou mesmo implementações para o MSX 2 e 2+.

Finalmente, segue uma pequena lista de programas gráficos que recomendo a todos que possuam um MSX 2 ou MSX 2+:

Dynamic Publisher
DD1 Graph (Dot Designer's Club)
Video Graphics Philips
Video Graphics Matsushita
Super Print
Graph Saurus
The Final Graph
The Sallas Free Hand
The MSX Paint 3
The Animator

Carlos Alberto Herszterg

• • •

Utilizo o MSX na área da advocacia há mais de três anos, com excelentes resultados. Além da edição de textos, utilizo banco de dados para o trato da clientela (cadastro, listagens, mala direta etc.), planilha de cálculos para as liquidações de sentenças, atualizações monetárias etc. e até edição gráfica, confeccionando cartão 1/8, papel timbrado, capas e cartazes. São cerca de 800 clientes e uma grande quantidade de serviço realizada em dois Experts (um DDPlus e um 2.0).

Assim, peço a gentileza de publicarem o meu nome e endereço, para aqueles que porventura se interessem na troca de informações ou mesmo para algum auxílio.

João de Oliveira Batista
Rua Moacy Avidos, 109/620
Praia do Canto - Vitória - ES
Cep 29057-230
Tel.: (027) 225-1341

TROCAS

Compro expansor de slots e interface RS-232. Quem quiser vender tais equipamentos entrar em contato com:

José Ayrton Egrafonte
Rua Anacleto Ramos, 92
Ferroviários
Cachoeiro de Itapemirim - ES
Cep 29308-020

• • •

Possuo um microcomputador MSX Expert DD Plus com drives de 3 1/2 e de 5 1/4 e cerca de 150 jogos. Gostaria que publicassem meu endereço para trocas de programas com usuários de todo o Brasil.

Israel Barbosa Sampaio
Rua Gialindo Baldiotti, 100
Jardim Guanabara - Monte Mor - SP
Cep 13190-000

• • •

Peço que publiquem o meu nome e endereço para trocas de programas e informações com outros usuários. Meu computador é um Expert Plus com drive 5 1/4.

Sandro Pinheiro de Souza Lemos
Rua 5 esq. com Rua 20, 1035/301
Centro - Goiânia - GO
Cep 74020-170

• • •

Gostaria de me corresponder com as pessoas que possuam um MSX 1.1 e um drive 5 1/4 (360 kB) para a troca de dicas e programas. Possuo mais de 300 programas para MSX, incluindo aplicativos, utilitários e principalmente jogos.

Responderei a todos que se corresponderem comigo.

Roberto g. Schwarz
Rua Telles, 216/303 - Centro
Pelotas - RS
Cep 96010-310
Tel: (0532) 2233-75

CLUBES

Nasce um clube totalmente diferente para o sistema MSX com muitos aplicativos, jogos, utilitários, revistas e livros. Sistema MSX. Podemos gravar em 5 1/4.

Os interessados devem remeter listagem e descrever a configuração do sistema.

MSX FRIENDS AND BROTHERS
Av. 7 de Setembro, 634/403
Centro - Divinópolis - MG
Cep 35.500-011

CONTRA
Vidas Infinitas

```
10 C$="CONTRA  "
20 BLOAD C$+".BIN"
30 DEFUSR=&HD1D3:FORA=0TO7
40 POKE&HD1DE,A*2:BLOADC$+".00"+
HEX$(A):GOSUB 60
50  X=USR(0):NEXT:DEFUSR=&HD1FC:
OUT&HD4,0:A=USR(0)
60 IFA=1THENPOKE&HBF30,0
70 RETURN
```

FAMICLE PARODIC
Vidas infinitas

```
10 C$="FAMICLE  "
20 BLOAD C$+".BIN"
30 DEFUSR=&HD1D3:FORA=0TO7
40 POKE&HD1DE,A*2:BLOADC$+".00"+
HEX$(A):GOSUB 60
50  X=USR(0):NEXT:DEFUSR=&HD1FC:
OUT&HD4,0:A=USR(0)
60 IFA=1THENPOKE&H9098,0
70 RETURN
```

Paulo Eduardo Costa Steinback

No próximo número

Sensacional!
Subdiretórios no MSX!

• • •

Memory Mapper
O esquema completo!

• • •

A solução do jogo
O Conde de Monte Cristo

• • •

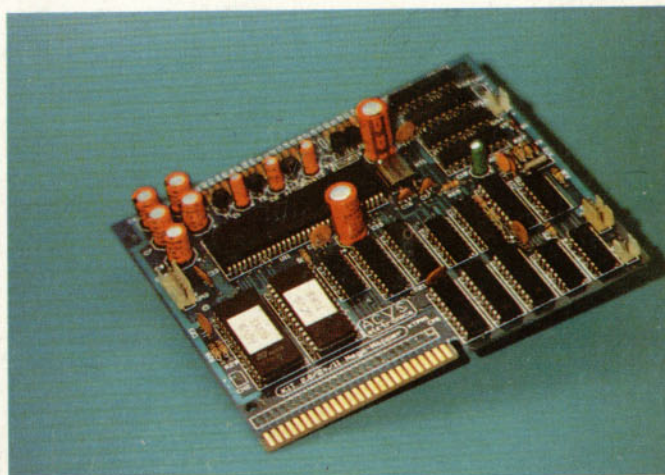
MSX 2 e MSX 2+, MSX Bits,
dicas e muito mais...

Imagine uma empresa que vende equipamentos, mantém atualizado o seu acervo de softwares, presta serviços de manutenção e ministra palestras e cursos em grandes empresas e faculdades.
Se você imagina tudo isso reunido num único lugar, você só pode estar pensando na **Focus Informática**.
Por esses entre outros motivos, como a criação da capa da revista que você acaba de ver, é que a Focus tornou-se o ponto de referência dos usuários de **Amiga**.

FOCUS Informática

R. Dona Inácia Uchôa, 135 - Vila Mariana - São Paulo - SP
CEP 04110-020 - Tel. (011) 549.7731

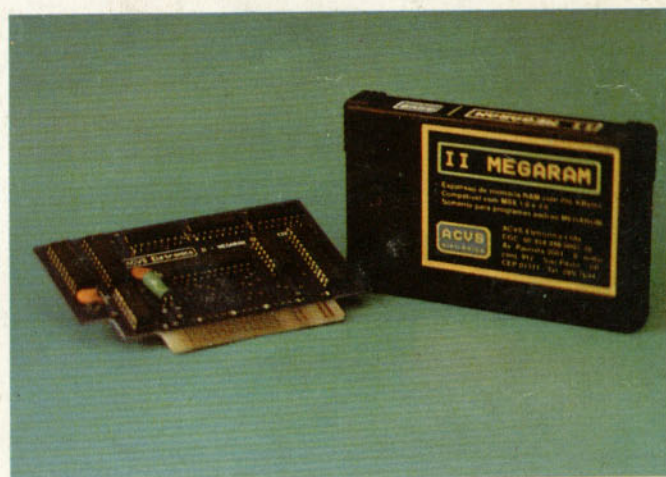
Transforme seu MSX em uma estação gráfica...

Tela digitalizada (foto em monitor RGB).

Placa eletrônica KIT 2+.

KIT 2+

• 19.268 cores • 256 KBytes RAM do usuário • 128 KBytes VRAM (vídeo) • 96 KBytes ROM-BASIC • TURBO-BASIC residente • 80 colunas de texto (mesmo pela TV) • Relógio/Calendário (mantido por bateria) • Movimentação fina das telas gráficas na horizontal e vertical • Resolução de 512 x 424 16 cores de 512

... e também em um Video-Game de alta resolução

Jogo SPACE-MANBOW (MEGAROM).

Placa e Cartucho II MEGARAM.

II-MEGARAM

• Expansão com 256 KBytes destinada a rodar os jogos MEGAROM gravados em disquetes. • Funciona em qualquer micro da Linha MSX. • Os jogos MEGAROM possuem alta definição gráfica e sonora.

Todos os produtos têm garantia de 1 ano.
KIT 2.0 e KIT 2+ são marcas registradas da ACVS Eletrônica Ltda.

Vectra eletrônica Representante Autorizado ACVS: Rua Antonio Gil, 1318 - sala 2 - Jardim Cupece - São Paulo - CEP: 04655-002 - Tel.: **(011) 564-3415**